

# Proxedo Network Security Suite 1.0 Log Messages Guide

Publication date June 04, 2020

## Abstract

This document describes the log messages of PNS





Copyright © 1996-2020 BalaSys IT Ltd.

This documentation and the product it describes are considered protected by copyright according to the applicable laws.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>). This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)

Linux™ is a registered trademark of Linus Torvalds.

Windows™ 10 is registered trademarks of Microsoft Corporation.

All other product names mentioned herein are the trademarks of their respective owners.

#### DISCLAIMER

is not responsible for any third-party websites mentioned in this document. does not endorse and is not responsible or liable for any content, advertising, products, or other material on or available from such sites or resources. will not be responsible or liable for any damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through any such sites or resources.



# Table of Contents

<b>Preface</b> .....	<b>xvii</b>
1. Single-line log message for connections .....	xvii
<b>1. Application-level Gateway log messages</b> .....	<b>1</b>
1.1. ....	1
1.1.1. Established connection .....	1
1.1.2. Error calling AuthenticationProvider method .....	1
1.1.3. Failed to parse URL filter database .....	1
1.1.4. Error opening socket .....	1
1.1.5. Error connecting UDP socket (nf) .....	1
1.1.6. Error querying local address (nf) .....	1
1.1.7. Error during setsockopt(IPPROTO_IP, IP_RECVORIGADDRS) .....	1
1.1.8. Error during setsockopt(IPPROTO_IPV6, IPV6_RECVORIGADDRS) .....	2
1.1.9. Cannot create socket .....	2
1.1.10. Dispatch thread starting .....	2
1.1.11. Accept queue stats .....	2
1.1.12. Dispatch thread exiting .....	2
1.1.13. Error creating dispatch thread, falling back to non-threaded mode .....	2
1.1.14. Incoming connection .....	2
1.1.15. Nobody was interested in this connection .....	3
1.1.16. Transparent listener connected directly, dropping connection .....	3
1.1.17. Error registering dispatch, previous entry specified accept_one .....	3
1.1.18. Internal error, dispatch entry not found (chain exists) .....	3
1.1.19. Internal error, dispatch entry not found (no chain) .....	3
1.1.20. Module successfully loaded .....	3
1.1.21. Module loading failed .....	3
1.1.22. eofmask updated .....	4
1.1.23. Packet stats timeout elapsed, and no timeout callback specified .....	4
1.1.24. Session ID not found for authorization .....	4
1.1.25. calling event .....	4
1.1.26. calling __destroy__() event .....	4
1.1.27. Internal error, trying to look up a non-ZProxyIface compatible interface .....	4
1.1.28. Error creating proxy thread .....	4
1.1.29. User initiated proxy termination request received .....	4
1.1.30. Trying verification without CRL check as directed by the policy .....	5
1.1.31. CRL has invalid nextUpdate field .....	5
1.1.32. CRL is expired .....	5
1.1.33. CRL not found for certificate .....	5
1.1.34. CRL not found for certificate .....	5
1.1.35. Error creating client socketpair for stacked proxy .....	5
1.1.36. Error creating server socketpair for stacked proxy .....	5
1.1.37. Stacking subproxy .....	6
1.1.38. Stacking subproxy .....	6
1.1.39. Error creating control socketpair for stacked proxy .....	6
1.1.40. Stacking program .....	6
1.1.41. Connecting to remote host .....	6



1.1.42. Connection denied by policy .....	6
1.1.43. Dispatcher on address .....	6
1.1.44. Error opening policy file .....	6
1.1.45. Error parsing policy file .....	7
1.1.46. Cannot find proxy module .....	7
1.1.47. Cannot find proxy module .....	7
1.1.48. Attribute fetched .....	7
1.1.49. Attribute changed .....	7
1.1.50. Error parsing satyr command .....	7
1.1.51. Internal error in z_policy_stream_new: input ZStream is NULL .....	7
1.1.52. Error connecting to satyrd .....	7
1.1.53. Satyr authentication failed, unable to determine target address .....	8
1.1.54. Satyr authentication failed, cannot connect to client .....	8
1.1.55. Error connecting to satyr, SSL handshake failed .....	8
1.1.56. Satyr authentication failed, unknown address family .....	8
1.1.57. Satyr authentication failed, unable to write request information .....	8
1.1.58. Satyr authentication failed, error reading answer .....	8
1.1.59. Invalid maximum aggregator, no source node .....	8
1.1.60. Invalid maximum aggregator, no source node .....	9
1.1.61. Failed to parse interval from node name .....	9
1.1.62. Invalid average aggregator, no source node .....	9
1.1.63. Failed to create SZIG socket .....	9
1.1.64. Error writing zas stream .....	9
1.1.65. Error booting & parsing policy .....	9
1.1.66. Error initializing policy .....	9
1.1.67. Error loading initial policy, exiting .....	9
1.1.68. Reloading policy .....	10
1.1.69. Error reloading policy, reverting to old .....	10
1.1.70. Unexpected EOF while transferring from server .....	10
1.1.71. I/O error opening error file .....	10
1.1.72. Eofmask is updated .....	10
1.1.73. Data transfer timed out .....	10
1.1.74. Data transfer interrupted by progress .....	10
1.1.75. Error reading request .....	11
1.1.76. Request details .....	11
1.1.77. Parse error, dropping request .....	11
1.1.78. Invalid character, dropping request .....	11
1.1.79. Username too long, dropping request .....	11
1.1.80. One hostname is too long in hostname chain, dropping request .....	11
1.1.81. Error write request .....	11
1.1.82. Error write request .....	11
1.1.83. Error write request .....	12
1.1.84. Error write request .....	12
1.1.85. Policy violation, abort session .....	12
1.1.86. Policy violation, drop session .....	12
1.1.87. fetching request .....	12
1.1.88. asking policy .....	12
1.1.89. connecting server .....	12



1.1.90. sending request .....	12
1.1.91. copying response .....	13
1.1.92. everything is done .....	13
1.1.93. Data connection accepted on client side .....	13
1.1.94. Data connection accepted on server side .....	13
1.1.95. Data connection established on client side .....	13
1.1.96. Connected to client, but connection is not expected .....	13
1.1.97. Data connection established on server side .....	13
1.1.98. Connected to server, but connection is not expected .....	13
1.1.99. Internal error, previous dispatcher not unregistered .....	14
1.1.100. Internal error, previous attach not unregistered .....	14
1.1.101. Internal error, previous attach not unregistered .....	14
1.1.102. Internal error, previous dispatcher not unregistered .....	14
1.1.103. Resetting data connection .....	14
1.1.104. Internal error, previous data connection is not closed properly .....	14
1.1.105. Internal error, cannot start data proxy because peers are not yet connected .....	14
1.1.106. Internal error, current command descriptor does not specify data transfer .....	15
1.1.107. Data connection established .....	15
1.1.108. Possible bounce attack .....	15
1.1.109. Possible bounce attack .....	15
1.1.110. Possible bounce attack .....	15
1.1.111. Possible bounce attack .....	15
1.1.112. Internal error, client side not connected .....	16
1.1.113. Internal error, server side not connected .....	16
1.1.114. Max_line_length above upper limit .....	16
1.1.115. Max_username_length above max_line_length .....	16
1.1.116. Max_password_length above max_line_length .....	16
1.1.117. Max_hostname_length above max_line_length .....	16
1.1.118. Error parsing valid_chars_username .....	16
1.1.119. Internal error in stream write, side is wrong .....	17
1.1.120. Cannot write full line .....	17
1.1.121. Server answer doesn't begin with number .....	17
1.1.122. Response arrived .....	17
1.1.123. Error reading from server .....	17
1.1.124. Line is too short to be a valid answer .....	17
1.1.125. Server answer has wrong continuation mark .....	17
1.1.126. Rejected answer .....	17
1.1.127. Error reading from client .....	18
1.1.128. Request fetched .....	18
1.1.129. Empty command. Aborting .....	18
1.1.130. Unknown command. Aborting .....	18
1.1.131. Internal error, known command but command parse is unset .....	18
1.1.132. This command not allowed in non-transparent mode .....	18
1.1.133. Request rejected .....	18
1.1.134. Proxy answer .....	18
1.1.135. Rejected command (aborting) .....	19
1.1.136. Bad policy type, aborting .....	19
1.1.137. Too many continuous lines in the answer .....	19



1.1.138. Reading from peer .....	19
1.1.139. Invalid port specified in non-transparent destination .....	19
1.1.140. Invalid character in username .....	19
1.1.141. Hostname specified in username is too long .....	19
1.1.142. Username too long .....	19
1.1.143. Username too long .....	20
1.1.144. Invalid character in username .....	20
1.1.145. Internal error, proxy in unknown state .....	20
1.1.146. Internal error, proxy in unknown state .....	20
1.1.147. Password too long .....	20
1.1.148. Password too long .....	20
1.1.149. Internal error, proxy in unknown state .....	20
1.1.150. Internal error, proxy in unknown state .....	21
1.1.151. Internal error, proxy in unknown state .....	21
1.1.152. Unexpected response to data transfer command .....	21
1.1.153. Internal error, proxy in unknown state .....	21
1.1.154. Invalid parameter for command .....	21
1.1.155. Invalid parameter for command .....	21
1.1.156. Internal error, proxy in unknown state .....	21
1.1.157. Missing parameter for the TYPE command .....	21
1.1.158. Valid, but unsupported transfer type specification .....	22
1.1.159. Unknown transfer type specification .....	22
1.1.160. Missing parameter to the MODE command .....	22
1.1.161. Invalid parameter to the MODE command .....	22
1.1.162. Missing parameter to the STRU command .....	22
1.1.163. Invalid parameter to the STRU command .....	22
1.1.164. Error preparing server-side data connection (PORT) .....	22
1.1.165. There was an error binding a server-side listener .....	23
1.1.166. Error parsing PASV response .....	23
1.1.167. Error preparing data connection to the server (PASV) .....	23
1.1.168. Error preparing server-side data connection listener (EPRT) .....	23
1.1.169. Cannot bind to the given address (EPRT) .....	23
1.1.170. Missing parameter (EPSV) .....	23
1.1.171. Bad parameter (EPSV), not beginning with bracket .....	23
1.1.172. Bad parameter (EPSV), not closing with bracket .....	24
1.1.173. Error parsing EPSV response .....	24
1.1.174. Bad parameter (EPSV), invalid port .....	24
1.1.175. Internal error, cannot detect server address .....	24
1.1.176. Unknown protocol type (EPSV) .....	24
1.1.177. Error preparing data connection to the server (EPSV) .....	24
1.1.178. Invalid parameters to the PORT command .....	24
1.1.179. Connection mode not supported .....	24
1.1.180. Error preparing client-side data connection (PORT-<;PASV) .....	25
1.1.181. Error parsing the server answer to the PASV command (PORT-<;PASV) .....	25
1.1.182. Error preparing client-side data connection (PORT) .....	25
1.1.183. Error parsing the server answer to the PORT command .....	25
1.1.184. Connection mode not supported .....	25
1.1.185. Error preparing client-side data connection listener (PASV) .....	25



1.1.186. Error preparing client-side data connection listener (PASV) .....	25
1.1.187. Error preparing client-side data connection listener (PASV-<;PORT) .....	26
1.1.188. Error preparing client-side data connection listener (PASV-<;PORT) .....	26
1.1.189. Error parsing the server answer to the PORT command (PASV-<;PORT) .....	26
1.1.190. Missing parameter (EPRT) .....	26
1.1.191. Bad parameter (EPRT) .....	26
1.1.192. Bad port parameter (EPRT) .....	26
1.1.193. Unknown protocol method (EPRT) .....	26
1.1.194. Bad host address (EPRT) .....	26
1.1.195. Connection mode not supported .....	27
1.1.196. Error preparing client connect (EPRT) .....	27
1.1.197. Bad server answer (EPRT) .....	27
1.1.198. Error preparing client connect (EPRT) .....	27
1.1.199. Connection mode not supported .....	27
1.1.200. Error preparing client listen (EPSV) .....	27
1.1.201. Error preparing client listen (EPSV) .....	27
1.1.202. Error preparing client listen (EPSV) .....	28
1.1.203. Error preparing client listen (EPSV) .....	28
1.1.204. Bad server answer (EPSV) .....	28
1.1.205. Error parsing command (ALLO) .....	28
1.1.206. Invalid stacking tuple returned by policy .....	28
1.1.207. Invalid stacking type .....	28
1.1.208. Invalid request, data transfer failed .....	28
1.1.209. Data transfer failed .....	28
1.1.210. Stacked proxy decision .....	29
1.1.211. Stacked proxy accepted data .....	29
1.1.212. Policy does not contain this request, using hard-coded default .....	29
1.1.213. Policy type invalid .....	29
1.1.214. Cannot parse policy line .....	29
1.1.215. Error in policy calling .....	29
1.1.216. Can't parsing return code .....	29
1.1.217. Policy does not contain this response, using hard-coded default .....	30
1.1.218. Answer type invalid .....	30
1.1.219. Bad policy line .....	30
1.1.220. Error in policy calling .....	30
1.1.221. Return code invalid from policy function .....	30
1.1.222. Policy does not contain this feature, dropping .....	30
1.1.223. Policy value invalid .....	30
1.1.224. Internal error, error code out of range .....	31
1.1.225. An error occurred, would serve error file, but silent mode is enabled .....	31
1.1.226. An error occurred, serving error file .....	31
1.1.227. Error writing stream, stream is NULL .....	31
1.1.228. Error writing stream .....	31
1.1.229. Only Basic authentication is supported .....	31
1.1.230. Invalid base64 encoded username:password pair .....	31
1.1.231. No colon is found in the decoded username:password pair .....	32
1.1.232. Invalid HTTP request received .....	32
1.1.233. Found category match for URL .....	32



1.1.234. CONNECT method without version specification .....	32
1.1.235. This request type is not permitted in transparent mode .....	32
1.1.236. No host header in transparent request, 'unknown' is used instead .....	32
1.1.237. 'Host:' header is required, and HTTP/0.9 can't transfer headers .....	32
1.1.238. No 'Host:' header in request, and policy requires this .....	32
1.1.239. Error parsing URL .....	33
1.1.240. Error parsing URL .....	33
1.1.241. Client attempted to use FTP over HTTP, which is currently disabled .....	33
1.1.242. Unsupported scheme in URL .....	33
1.1.243. Too long hostname in URL .....	33
1.1.244. Accounting .....	33
1.1.245. Invalid item in request hash .....	33
1.1.246. Error parsing HTTP_REQ_POLICY tuple in request hash .....	34
1.1.247. Error parsing HTTP_REQ_REJECT in request hash .....	34
1.1.248. Unknown request hash item .....	34
1.1.249. Policy requested to send custom response .....	34
1.1.250. Request not permitted by policy .....	34
1.1.251. Request not permitted by policy .....	34
1.1.252. Connecting to this port is prohibited by policy .....	34
1.1.253. Internal error initializing server stream .....	35
1.1.254. Error in HTTP/0.9 compatibility code, line buffer full .....	35
1.1.255. Server falled back to HTTP/0.9 which is prohibited by policy .....	35
1.1.256. Invalid HTTP response heading .....	35
1.1.257. Error parsing response version .....	35
1.1.258. Unsupported protocol version .....	35
1.1.259. Invalid response hash item .....	35
1.1.260. Error parsing HTTP_RSP_POLICY in response hash .....	36
1.1.261. Error parsing HTTP_RSP_REJECT in response hash .....	36
1.1.262. Invalid response hash item .....	36
1.1.263. Response not permitted by policy .....	36
1.1.264. Missing port number in CONNECT request .....	36
1.1.265. Parent proxy refused our CONNECT request .....	36
1.1.266. Starting connect method .....	36
1.1.267. Internal error, connectMethod is expected to return a proxy instance .....	37
1.1.268. Fetching request and headers .....	37
1.1.269. processing request and headers .....	37
1.1.270. Maximum keep-alive request reached, setting connection mode to close .....	37
1.1.271. Filtering request and headers .....	37
1.1.272. Reprocessing filtered request .....	37
1.1.273. Sending request and headers, copying request data .....	37
1.1.274. Fetching response and headers .....	37
1.1.275. Processing response and headers .....	38
1.1.276. Filtering response and headers .....	38
1.1.277. Copying response and headers, copying response data .....	38
1.1.278. Internal error, invalid server_protocol .....	38
1.1.279. exiting keep-alive loop .....	38
1.1.280. Chunk length line too long .....	38
1.1.281. Invalid chunk length .....	38





1.1.282. Invalid chunk length .....	38
1.1.283. Chunk too large .....	39
1.1.284. Unexpected EOF while dechunking stream .....	39
1.1.285. Chunk footer is not an empty line .....	39
1.1.286. Stacked proxy sent a content-length hint, using it .....	39
1.1.287. Invalid stacking tuple returned by policy .....	39
1.1.288. Invalid stacking type .....	39
1.1.289. The header 'Content-Length' was present, but is not a number .....	39
1.1.290. Invalid request, data transfer failed .....	40
1.1.291. Data transfer failed .....	40
1.1.292. Stacked proxy decision .....	40
1.1.293. Stacked proxy accepted data .....	40
1.1.294. Invalid FTP response, line too short .....	40
1.1.295. Invalid FTP response, response code not numeric .....	40
1.1.296. Invalid FTP response, continuation line contains different status code .....	40
1.1.297. FTP server returned an unhandled status code for the login request .....	40
1.1.298. Error logging in .....	41
1.1.299. Error in FTP greeting, expecting 220 .....	41
1.1.300. Error changing directory on server .....	41
1.1.301. Error changing directory on server .....	41
1.1.302. Error changing directory on server .....	41
1.1.303. Error sending RETR or LIST command to FTP server .....	41
1.1.304. Error establishing data connection to FTP server .....	41
1.1.305. Error reading data channel confirmation from FTP server .....	41
1.1.306. Error reading 226 from FTP server .....	42
1.1.307. Invalid HTTP_HDR_CHANGE_NAME rule in header processing .....	42
1.1.308. Invalid HTTP_HDR_CHANGE_VALUE rule in header processing .....	42
1.1.309. Invalid HTTP_HDR_CHANGE_BOTH rule in header processing .....	42
1.1.310. Invalid value in header action tuple .....	42
1.1.311. Error reading from peer while fetching headers .....	42
1.1.312. First header starts with white space .....	42
1.1.313. Invalid HTTP header .....	43
1.1.314. Too many header lines .....	43
1.1.315. Request method empty, or too long .....	43
1.1.316. URL missing, or too long .....	43
1.1.317. Protocol version missing, or too long .....	43
1.1.318. Request details .....	43
1.1.319. Invalid HTTP status line .....	43
1.1.320. Response version empty or too long .....	43
1.1.321. Response code empty or too long .....	44
1.1.322. Response details .....	44
1.1.323. Unknown protocol version .....	44
1.1.324. Unknown protocol version .....	44
1.1.325. Invalid return value of packetStats(), integer required .....	44
1.1.326. packetStats() requested to abort session .....	44
1.1.327. Response line too long .....	44
1.1.328. Response status is invalid .....	44
1.1.329. Response fetched with parameter .....	45



1.1.330. Response fetched	45
1.1.331. Data transfer failed	45
1.1.332. Request line too long	45
1.1.333. Request fetched with parameter	45
1.1.334. Request fetched	45
1.1.335. Unknown request command	45
1.1.336. Request command not allowed in this state	46
1.1.337. Dropping request parameter, no parameter allowed	46
1.1.338. The required numerical parameter of the request is missing	46
1.1.339. The numerical parameter of the request is not in the given range	46
1.1.340. The numerical parameter of the request is negative	46
1.1.341. The numerical parameter of the request is zero	46
1.1.342. The numerical parameter of the request contains junk after the number	46
1.1.343. The first numerical parameter of the request is not in the given range	47
1.1.344. The first numerical parameter of the request is negative	47
1.1.345. Only one numerical argument in request	47
1.1.346. The second numerical parameter of the request is not in the given range	47
1.1.347. The second numerical parameter of the request is a negative number	47
1.1.348. The numerical parameter of the request contain junk after the number	47
1.1.349. Username is too long	47
1.1.350. Password is too long	48
1.1.351. The username parameter is too long or the digest parameter is missing	48
1.1.352. Error parsing the MD5 digest	48
1.1.353. The required numerical parameter of the response is missing	48
1.1.354. The numerical parameter of the response is not in the given range	48
1.1.355. The numerical parameter of the response is a negative number	48
1.1.356. The numerical parameter of the response contains junk after the number	48
1.1.357. The numerical parameter of the response is not in the given range	49
1.1.358. The required numerical parameter of the response is missing	49
1.1.359. The numerical parameter of the response is a negative number	49
1.1.360. The second numerical parameter of the response is not in the given range	49
1.1.361. The required second numerical parameter of the response is missing	49
1.1.362. The second numerical parameter of the response is a negative number	49
1.1.363. The second numerical parameter of the response contains junk after the number	49
1.1.364. Stacked proxy rejected contents	50
1.1.365. Policy does not contain this request, using hard-coded default	50
1.1.366. Policy type is invalid	50
1.1.367. Cannot parse policy line	50
1.1.368. Error in policy call	50
1.1.369. Cannot parse the return code	50
1.1.370. Policy does not contain this request, using hard-coded default	50
1.1.371. Policy type is invalid	51
1.1.372. Cannot parse policy line	51
1.1.373. Error in policy call	51
1.1.374. Cannot parse return code	51
1.1.375. Stack policy type is invalid	51
1.1.376. Cannot parse stack policy line	51
1.1.377. Cannot parse stack policy line	51



1.1.378. Error in policy call .....	52
1.1.379. Cannot parse return code .....	52
1.1.380. Invalid command verb in request .....	52
1.1.381. Request parsed .....	52
1.1.382. Fetching request .....	52
1.1.383. Request line too long .....	52
1.1.384. Processing request .....	52
1.1.385. Command not permitted in this state .....	52
1.1.386. Unknown command .....	53
1.1.387. Copying request to server .....	53
1.1.388. Error sending request .....	53
1.1.389. Fetching authentication request .....	53
1.1.390. Auth request line too long .....	53
1.1.391. Copying authentication request to server .....	53
1.1.392. Error sending authentication request .....	53
1.1.393. SMTP reply contains non-numeric characters .....	54
1.1.394. Invalid continuation character .....	54
1.1.395. Fetching response .....	54
1.1.396. Response line too long .....	54
1.1.397. Invalid SMTP reply, reply code changed .....	54
1.1.398. Response parsed .....	54
1.1.399. Processing response .....	54
1.1.400. Copying response to client .....	54
1.1.401. Error sending SMTP reply .....	55
1.1.402. Invalid policy ignored, allowing 250 response to NOOP is required .....	55
1.1.403. Invalid policy ignored, allowing 354 response to DATA is required .....	55
1.1.404. Invalid contents .....	55
1.1.405. Error occurred while scanning contents .....	55
1.1.406. Rejecting empty message .....	55
1.1.407. Error sending RSET to the server .....	55
1.1.408. Exiting SMTP loop .....	55
1.1.409. Error parsing local part .....	56
1.1.410. Local part does not end in '@' .....	56
1.1.411. Invalid domain name in path .....	56
1.1.412. Path does not start with '>..' .....	56
1.1.413. Invalid source route information .....	56
1.1.414. Invalid address information .....	56
1.1.415. Path does not begin with '>..' but ends with '<..' .....	56
1.1.416. Path begins with '>..' but does not end with '<..' .....	56
1.1.417. Invalid SIZE extension in the MAIL command .....	57
1.1.418. Invalid BODY extension in the MAIL command .....	57
1.1.419. Invalid AUTH sender, not an xtext .....	57
1.1.420. Invalid extension in the MAIL command .....	57
1.1.421. Invalid request policy type .....	57
1.1.422. Error in request policy .....	57
1.1.423. Error in request policy .....	57
1.1.424. The verdict returned by the policy is not an int .....	58
1.1.425. Invalid response policy .....	58



1.1.426. Error in response policy .....	58
1.1.427. Error in response policy .....	58
1.1.428. The verdict returned by the policy is not an int .....	58
1.1.429. Buffer overflow during protocol decoding, aborting session .....	58
1.1.430. Internal error, not enough literals .....	58
1.1.431. Internal error, too many literals .....	59
1.1.432. Error while writing policy response .....	59
1.1.433. Error while writing policy response .....	59
1.1.434. Failed to parse command .....	59
1.1.435. Error writing to client .....	59
1.1.436. Reconstructed request .....	59
1.1.437. Illegal tag .....	59
1.1.438. Duplicate tag .....	59
1.1.439. Command not allowed in this state .....	60
1.1.440. Error parsing command .....	60
1.1.441. Request denied by policy .....	60
1.1.442. Dropping request .....	60
1.1.443. Policy responds to client .....	60
1.1.444. Aborting connection due to request .....	60
1.1.445. Too many pending requests .....	60
1.1.446. Error writing to server .....	60
1.1.447. Stray continuation response .....	61
1.1.448. Error writing to server .....	61
1.1.449. Error writing to server .....	61
1.1.450. Error writing to server .....	61
1.1.451. Server returned untagged line during authentication exchange, dropping it .....	61
1.1.452. Aborting connection due to a response .....	61
1.1.453. Dropping response .....	61
1.1.454. Invalid response .....	61
1.1.455. Error writing to client .....	62
1.1.456. Response arrived for unknown request .....	62
1.1.457. Aborting connection due to a response .....	62
1.1.458. Dropping response .....	62
1.1.459. Invalid response .....	62
1.1.460. Error writing to client .....	62
1.1.461. Complete response arrived .....	62
1.1.462. Reconstructed response .....	62
1.1.463. Reading literal .....	63
1.1.464. No free literal space .....	63
1.1.465. Illegal character .....	63
1.1.466. Illegal literal length form .....	63
1.1.467. Literal length attribute not closed .....	63
1.1.468. Stacked proxy rejected contents .....	63
1.1.469. Policy type is invalid .....	63
1.1.470. Cannot parse stack policy line .....	63
1.1.471. Cannot parse stack policy line .....	64
1.1.472. Error in policy call .....	64
1.1.473. Cannot parse return code .....	64



1.1.474. Response count for the request reached limit, drop packets .....	64
1.1.475. Response count for request reached limit, aborting connection .....	64
1.1.476. Invalid value for response_overrun_action, falling back to LDAP_RSP_DROP .....	64
1.1.477. Policy nesting do deep .....	64
1.1.478. Policy type invalid .....	65
1.1.479. Cannot parse policy line .....	65
1.1.480. Cannot parse policy line .....	65
1.1.481. Cannot parse policy line .....	65
1.1.482. Cannot parse policy line .....	65
1.1.483. Error in policy calling .....	65
1.1.484. Policy does not contain this header, using hard-coded default .....	65
1.1.485. Policy does not contain this mime-type, using hard-coded default .....	65
1.1.486. Answer type invalid .....	66
1.1.487. Bad policy line .....	66
1.1.488. Error in policy calling .....	66
1.1.489. Return code invalid from policy function .....	66
1.1.490. Policy does not contain this error, using hard-coded default .....	66
1.1.491. Cannot parse policy line .....	66
1.1.492. Error in policy calling .....	66
1.1.493. Error parsing return code .....	66
1.1.494. Stack policy type is invalid .....	67
1.1.495. Cannot parse stack policy line .....	67
1.1.496. Error in policy call .....	67
1.1.497. Cannot parse return code .....	67
1.1.498. Stacked proxy rejected contents .....	67
1.1.499. Invalid stacking type .....	67
1.1.500. Invalid stacking tuple returned by policy .....	67
1.1.501. Evaluating BAACL rule .....	68
1.1.502. BAACL result .....	68
1.1.503. Cache over shift-threshold, shifting .....	68
1.1.504. Server connection failure .....	68
1.1.505. Server connection established .....	68
1.1.506. Server connection failure, no destination .....	68
1.1.507. All destinations are down, clearing cache and trying again .....	68
1.1.508. Destination is down, skipping .....	68
1.1.509. Destination is down, keeping state .....	69
1.1.510. Side stacking failed, socketPair failed .....	69
1.1.511. Side-stacking proxy instance .....	69
1.1.512. Side-stacking failed .....	69
1.1.513. Unable to determine service, KZorp service lookup failed .....	69
1.1.514. No applicable service found for this client zone (cached) .....	69
1.1.515. No applicable service found for this client zone .....	69
1.1.516. No applicable service found for this client & server zone (cached) .....	70
1.1.517. No applicable service found for this client & server zone .....	70
1.1.518. Matching regexp found .....	70
1.1.519. Error opening match file .....	70
1.1.520. Error opening ignore file .....	70
1.1.521. Cached recipient match found .....	70



1.1.522. Recipient validity not cached, trying the direct way .....	70
1.1.523. SMTP sender was rejected, unable to verify user existence .....	71
1.1.524. Server accepted recipient .....	71
1.1.525. Server rejected recipient .....	71
1.1.526. SMTP error during recipient validity checking .....	71
1.1.527. Before NAT mapping .....	71
1.1.528. After NAT mapping .....	71
1.1.529. Proxy ending .....	71
1.1.530. Proxy destroy .....	71
1.1.531. Error while stacking child proxy .....	72
1.1.532. Stacking custom child .....	72
1.1.533. User authentication successful .....	72
1.1.534. Starting proxy instance .....	72
1.1.535. Ending proxy instance .....	72
1.1.536. Outbound connection not permitted .....	72
1.1.537. Error resolving hostname .....	72
1.1.538. Instance definition not found in policy .....	72
1.1.539. Error downloading KZorp configuration, Python traceback follows .....	73
1.1.540. Deinitialization requested for instance .....	73
1.1.541. Cleaning up instance .....	73
1.1.542. Error flushing KZorp configuration .....	73
1.1.543. Client foreign port below 1024 .....	73
1.1.544. Server foreign port below 1024 in passive mode .....	73
1.1.545. Server foreign port is not good in active mode .....	73
1.1.546. Unknown side when calling bounceCheck .....	74
1.1.547. Http accounting .....	74
1.1.548. Request administratively prohibited .....	74
1.1.549. Sender address administratively prohibited .....	74
1.1.550. Sender check successful .....	74
1.1.551. Forbidden percent found in address local-part .....	74
1.1.552. Forbidden exclamation mark found in address local-part .....	74
1.1.553. Relaying denied .....	74
1.1.554. Relay check successful .....	75
1.1.555. Recipient address administratively prohibited .....	75
1.1.556. Recipient check successful .....	75
1.1.557. Relay check, checking client_zone .....	75
1.1.558. Relay check, checking mail domain .....	75
1.1.559. Signal received, stackdump follows .....	75
1.1.560. Invalid license file .....	75
1.1.561. Error initializing Python policy engine .....	75
<b>2. Management Server log messages .....</b>	<b>77</b>
2.1. ....	77
2.1.1. Module successfully loaded .....	77
2.1.2. Module loading failed .....	77
2.1.3. Module loading failed .....	77
2.1.4. Module loading failed .....	77
2.1.5. Module loading failed .....	77
2.1.6. Cannot open plugin directory .....	77



2.1.7. Module loading failed .....	77
2.1.8. Cannot open plugin directory .....	78
2.1.9. Signal received, stackdump follows .....	78
2.1.10. SIGHUP received, reloading config .....	78
<b>3. MS Transfer Agent log messages .....</b>	<b>79</b>
3.1. ....	79
3.1.1. Signal received, stackdump follows .....	79
<b>4. Authentication Server log messages .....</b>	<b>80</b>
4.1. ....	80
4.1.1. Signal received, stackdump follows .....	80
4.1.2. Module successfully loaded .....	80
4.1.3. Module initialization failed .....	80
4.1.4. Error dlopening module .....	80
4.1.5. Cannot open plugin directory .....	80
<b>5. Authentication Agent log messages .....</b>	<b>81</b>
5.1. ....	81
5.1.1. Signal received, stackdump follows .....	81
5.1.2. Signal received, stackdump follows .....	81
<b>6. Content Filtering log messages .....</b>	<b>82</b>
6.1. ....	82
6.1.1. Module successfully loaded .....	82
6.1.2. Module initialization failed .....	82
6.1.3. Error dlopening module .....	82
6.1.4. Cannot open plugin directory .....	82
6.1.5. Signal received, stackdump follows .....	82
<b>Appendix A. Creative Commons Attribution Non-commercial No Derivatives (by-nc-nd) License</b> .....	<b>83</b>



# List of Procedures

1. Single-line log message for connections .....	xvii
--	------





- *client proto*: The transport protocol used in the client-side connection. This is the protocol used in the transport layer (Layer 4) of the OSI model (for example, TCP, UDP, ICMP, and so on).
- *client IP*: The IP address of the client.
- *client port*: The port number of the client.
- *client zone*: The zone the client belongs to.
- *server proto*: The transport protocol used in the server-side connection. This is the protocol used in the transport layer (Layer 4) of the OSI model (for example, TCP, UDP, ICMP, and so on).
- *server IP*: The IP address of the server connected by PNS.
- *server port*: The port number of the server connected by PNS.
- *server zone*: The zone the client belongs to.
- *client local IP address (after NAT)*: The IP address of PNS used in the client-side connection.
- *client local port (after NAT)*: The port number of PNS used in the client-side connection.
- *server local IP address (after NAT)*: The IP address of PNS used in the server-side connection.
- *server local port (after NAT)*: The port number of PNS used in the server-side connection.
- *verdict*: Indicates what PNS decided about the connection.
  - *ACCEPTED*: PNS accepted the connection, and it was established without any problems.
  - *DENIED\_BY\_CONNECTION\_FAIL*: Connection failed, that is, it was allowed to pass PNS but timed out on the server.
  - *DENIED\_BY\_LIMIT*: PNS rejected the connection because it exceeded the **Thread limit** parameter of the instance, or the **Limit concurrency** parameter of the service.
  - *DENIED\_BY\_POLICY*: PNS did not find a matching firewall rule for the connection.
  - *DENIED\_BY\_UNKNOWN\_FAIL*: The connection failed for some reason.
  - *NO\_SERVICE\_FOUND*: PNS did not find a matching service for the parameters of the connection.
- *info*: Additional information about the connection (if any).

```
core.summary(4): (svc/example_service_name:1234): Connection summary; rule_id='N/A'  
  session_start='1406290229', session_end='1406290229',  
client_proto='TCP', client_address='10.10.1.10', client_port='3394',  
client_zone='example-zone',  
server_proto='TCP', server_address='10.10.1.10', server_port='3394',  
server_zone='example-zone',  
client_local='10.10.1.10', client_local_port='55268',  
server_local='10.10.60.253', server_local_port='55258',  
verdict='ACCEPTED',  
info='Ending forwarded session'
```

```
core.summary(4): (svc/example_service_name:1234): Connection summary; rule_id='N/A'  
  session_start='1406290229', session_end='1406290229',  
client_proto='TCP', client_address='10.10.1.10', client_port='3394',  
client_zone='example-zone',  
server_proto='TCP', server_address='10.10.1.10', server_port='3394',  
server_zone='example-zone',  
client_local='10.10.1.10', client_local_port='55268',
```



```
server_local='10.10.60.253', server_local_port='55258',  
verdict='NO_SERVICE_FOUND',  
info='No applicable service found for this client & server zone, dropping packet'
```

### Steps:

- Step 1. Login to your PNS host.
- Step 2. Execute the following commands:

```
echo 1 > /proc/sys/net/netfilter/kzorp/log_session_verdict  
zorpctl log --logspec 'core:4'
```

- Step 3. Repeat this procedure on your other PNS firewall hosts.

### Expected result:

When a connection ends, PNS logs a single-line log message about the connection, for example:

```
core.summary(4): (svc/example_service_name:1234): Connection summary;  
rule_id='N/A' session_start='1406290229', session_end='1406290229',  
client_proto='TCP', client_address='10.10.1.10', client_port='3394',  
client_zone='example-zone',  
server_proto='TCP', server_address='10.10.1.10', server_port='3394',  
server_zone='example-zone',  
client_local='10.10.1.10', client_local_port='55268',  
server_local='10.10.60.253', server_local_port='55258',  
verdict='REJECTED_BY_POLICY'  
info=''
```



# Chapter 1. Application-level Gateway log messages

## 1.1.1. Established connection

**core.debug 6**

This message reports that the connection was successfully established.

## 1.1.2. Error calling AuthenticationProvider method

**core.error 1**

This message indicates that the given python function is not callable. Check your AuthenticationProvider methods. @see: Auth.AuthenticationProvider

## 1.1.3. Failed to parse URL filter database

**core.error 1**

This message indicates that the url filter database can't be loaded. This could be because of a bad file format, version mismatch or an error in the generated database.

## 1.1.4. Error opening socket

**core.error 3**

This message indicate that Zorp failed opening a new socket. It is likely that Zorp reached some resource limit.

## 1.1.5. Error connecting UDP socket (nf)

**core.error 3**

This message indicates that UDP connection failed.

## 1.1.6. Error querying local address (nf)

**core.error 3**

This message indicates that Zorp was unable to query the local address.

## 1.1.7. Error during setsockopt(IPPROTO\_IP, IP\_RECVORIGADDRS)

**core.error 3**



This message indicates that the setsockopt requesting reception of original destination addresses of UDP frames failed.

### **1.1.8. Error during setsockopt(IPPROTO\_IPV6, IPV6\_RECVORIGADDRS)**

#### **core.error 3**

This message indicates that the setsockopt requesting reception of original destination addresses of UDP frames failed.

### **1.1.9. Cannot create socket**

#### **core.error 2**

This message indicate that the creation of a new socket failed for the given reason. It is likely that the system is running low on memory, or the system is running out of the available fds.

### **1.1.10. Dispatch thread starting**

#### **core.debug 4**

This message reports that a new dispatcher thread is starting. This is used if threaded dispatching is enabled.  
@see: Dispatcher

### **1.1.11. Accept queue stats**

#### **core.debug 4**

This message reports the dispatcher average accept queue length status.

### **1.1.12. Dispatch thread exiting**

#### **core.debug 4**

This message reports that the dispatcher thread is exiting. It is likely that Zorp unbinds from that address.

### **1.1.13. Error creating dispatch thread, falling back to non-threaded mode**

#### **core.error 2**

This message indicates that Zorp was unable to create a dispatcher thread for accepting new connection, and it is reverting back to the original non-threaded mode. It is likely that Zorp reached its thread or resource limit. Check your logs for further information.

### **1.1.14. Incoming connection**

#### **core.debug 6**

This message reports that a new connections is coming.



### 1.1.15. Nobody was interested in this connection

#### core.error 3

This message indicates that a new connection was accepted, but no Dispatcher/Proxy was interested in it.

### 1.1.16. Transparent listener connected directly, dropping connection

#### core.error 1

This message indicates that Dispatcher was configured to be accept transparent connections, but it was connected directly. Configure it either non-transparent or deny direct access to it and set up the appropriate TPROXY rule. @see: Dispatcher

### 1.1.17. Error registering dispatch, previous entry specified accept\_one

#### core.error 1

This message indicates that a Dispatcher/Proxy was unable bind to a specified address, because another instance is already listening there and specified that only one connection could be accepted.

### 1.1.18. Internal error, dispatch entry not found (chain exists)

#### core.error 1

This message indicates that a Dispatcher/Proxy tries to unbind from the specified address, but have not registered itself to that address.

### 1.1.19. Internal error, dispatch entry not found (no chain)

#### core.error 1

This message indicates that a Dispatcher/Proxy tries to unbind from the specified address, but Zorp does not bind to that address.

### 1.1.20. Module successfully loaded

#### core.debug 8

This message serves informational purposes, and indicates that the given module was successfully loaded from the given shared object.

### 1.1.21. Module loading failed

#### core.error 1

This message indicates that loading a proxy module failed.

### 1.1.22. eofmask updated

**core.debug 7**

This message reports that the end-of-file status has been updated.

### 1.1.23. Packet stats timeout elapsed, and no timeout callback specified

**core.error 3**

This message indicates that packet stats interval was specified, but no action was configured to handle the event. Check your policy for packetStats event.

### 1.1.24. Session ID not found for authorization

**core.info 3**

Indicates that an invalid session ID is specified by the authorize command

### 1.1.25. calling event

**core.debug 7**

This message reports that Zorp is about to call the proxy's %event() event.

### 1.1.26. calling \_\_destroy\_\_() event

**core.debug 7**

This message reports that Zorp is about to call the proxy's \_\_destroy\_\_() event. This method handles the pre destroy tasks, like the shutdown of the server side connection.

### 1.1.27. Internal error, trying to look up a non-ZProxyIface compatible interface

**core.error 3**

This message indicates an internal error, please contact your Zorp support for assistance.

### 1.1.28. Error creating proxy thread

**core.error 2**

This message indicates that Zorp was unable to create a new thread for the new proxy instance. It is likely that Zorp reached a thread limit, or not enough resource is available.

### 1.1.29. User initiated proxy termination request received

**core.info 2**



A stop request arrived to the proxy. It has to be stopped.

### **1.1.30. Trying verification without CRL check as directed by the policy**

**core.policy 5**

chain was not found during verification but the proxy configuration explicitly allows missing CRLs.

### **1.1.31. CRL has invalid nextUpdate field**

**core.error 1**

This message indicates an invalid Certificate Revocation List (CRL), because it has an invalid nextUpdate field.

### **1.1.32. CRL is expired**

**core.error 1**

This message indicates an invalid Certificate Revocation List (CRL), because it is expired.

### **1.1.33. CRL not found for certificate**

**core.error 1**

This message indicates that no certificate revocation list was found for a CA certificate, and Zorp configuration requires that a CRL must be present.

### **1.1.34. CRL not found for certificate**

**core.error 1**

This message indicates that no certificate revocation list was found for a CA certificate, and Zorp configuration requires that a CRL must be present.

### **1.1.35. Error creating client socketpair for stacked proxy**

**core.error 1**

This message indicates that stacking a child proxy failed, because creating an AF\_UNIX domain socketpair failed on the client side.

### **1.1.36. Error creating server socketpair for stacked proxy**

**core.error 1**

This message indicates that stacking a child proxy failed, because creating an AF\_UNIX domain socketpair failed on the server side.



### **1.1.37. Stacking subproxy**

**core.debug 6**

This message reports that Zorp is about to stack a proxy class with the given fds as communication channels.

### **1.1.38. Stacking subproxy**

**core.debug 6**

This message reports that Zorp is about to stack a proxy class with the given fds as communication channels.

### **1.1.39. Error creating control socketpair for stacked proxy**

**core.error 1**

This message indicates that stacking a child proxy failed, because creating an AF\_UNIX domain socketpair failed for the control channel.

### **1.1.40. Stacking program**

**core.debug 6**

This message reports that Zorp is about to stack a program with the given fds as communication channels.

### **1.1.41. Connecting to remote host**

**core.debug 7**

This message indicates that Zorp began establishing connection with the indicated remote host.

### **1.1.42. Connection denied by policy**

**core.policy 1**

This message indicates that the decision layer denied the given connection.

### **1.1.43. Dispatcher on address**

**core.debug 7**

This message indicates that a Dispatcher on the given local address is started.

### **1.1.44. Error opening policy file**

**core.error 0**

This message indicates that Zorp was unable to open the policy file. Check the permissions of your policy file.



### 1.1.45. Error parsing policy file

#### core.error 0

This message indicates that Zorp was unable to parse the policy file. Check the logs for further information on where the error occurred in the policy.

### 1.1.46. Cannot find proxy module

#### core.error 1

This message indicates that Zorp was unable to find the required proxy module. Check your installation, or contact your Zorp support for assistance.

### 1.1.47. Cannot find proxy module

#### core.error 1

This message indicates that Zorp was unable to find the required proxy module. Check your installation, or contact your Zorp support for assistance.

### 1.1.48. Attribute fetched

#### core.debug 6

This message reports that the given proxy-exported attribute was fetched, and it contained the also given value.

### 1.1.49. Attribute changed

#### core.debug 6

This message reports that the given proxy-exported attribute was changed to the given value.

### 1.1.50. Error parsing satyr command

#### core.error 2

This message indicates that the Satyr authentication client sent an invalid command. Check your Satyr package version, or contact your Zorp support for assistance.

### 1.1.51. Internal error in z\_policy\_stream\_new: input ZStream is NULL

#### core.error 3

This message indicates an internal error, please contact your Zorp support for assistance.

### 1.1.52. Error connecting to satyrd

#### satyr.error 3



This message indicates that establishing a connection to the authentication agent failed for the given reason.

### **1.1.53. Satyr authentication failed, unable to determine target address**

#### **core.error 3**

This message indicates that the SATYR authentication subsystem does not support the address family the client used to access the service to be authenticated. Please contact your Zorp support for assistance.

### **1.1.54. Satyr authentication failed, cannot connect to client**

#### **core.error 3**

This message indicates that Zorp was unable to connect to the client Satyr authentication client. It is likely that the client does not run Satyr on the default port or something blocks the connection. Check your client configuration.

### **1.1.55. Error connecting to satyr, SSL handshake failed**

#### **core.error 3**

This message indicates that the SSL handshake with the authentication agent failed because of the given SSL error condition.

### **1.1.56. Satyr authentication failed, unknown address family**

#### **core.error 3**

This message indicates that Satyr authentication failed, because non-IPv4 address was used. Please contact your Zorp support for assistance.

### **1.1.57. Satyr authentication failed, unable to write request information**

#### **core.error 3**

This message indicates that Satyr authentication failed, because Zorp was unable to send data to Satyr.

### **1.1.58. Satyr authentication failed, error reading answer**

#### **core.error 3**

This message indicates that Satyr authentication failed, because Zorp was unable to read the client authentication answer.

### **1.1.59. Invalid maximum aggregator, no source node**

#### **core.error 3**

This message indicates an internal error, please contact your Zorp support for assistance.



### **1.1.60. Invalid maximum aggregator, no source node**

**core.error 3**

This message indicates an internal error, please contact your Zorp support for assistance.

### **1.1.61. Failed to parse interval from node name**

**core.error 3**

This message indicates an internal error, please contact your Zorp support for assistance.

### **1.1.62. Invalid average aggregator, no source node**

**core.error 3**

This message indicates an internal error, please contact your Zorp support for assistance.

### **1.1.63. Failed to create SZIG socket**

**core.info 4**

This message reports that the SZIG framework was unable to create its socket and thus zorpctl won't be able to access and display internal Zorp information.

### **1.1.64. Error writing zas stream**

**core.error 3**

This message indicates that communication was interrupted to ZAS.

### **1.1.65. Error booting & parsing policy**

**core.error 0**

This message indicates that Zorp was unable to load the policy. It is likely that the policy has any kind of syntactical problem. Check the traceback in the log to find out where the problem occurs.

### **1.1.66. Error initializing policy**

**core.error 0**

This message indicates that Zorp was unable to initialize the policy.

### **1.1.67. Error loading initial policy, exiting**

**core.error 0**



This message indicates that the loading of the initial policy failed, because of some policy problem. Check the log to find out where the problem occurs.

### **1.1.68. Reloading policy**

#### **core.info 0**

This message reports that Zorp caught a HUP signal and tries to reload its policy.

### **1.1.69. Error reloading policy, reverting to old**

#### **core.error 0**

This message indicates that Zorp was unable to load the new policy, and reverts to the old one. Check the logs to find out where the error occurs in the new policy.

### **1.1.70. Unexpected EOF while transferring from server**

#### **core.error 4**

This message indicates that server unexpectedly closed its connection.

### **1.1.71. I/O error opening error file**

#### **core.error 3**

This message indicates that Zorp was unable to open the error file for the given reason. It is likely that the file does not exist, or has too restrictive permissions.

### **1.1.72. Eofmask is updated**

#### **core.debug 7**

This message reports that the data-transfer to or from some endpoint is closed.

### **1.1.73. Data transfer timed out**

#### **core.error 3**

This message indicates the data transfer timed out.

### **1.1.74. Data transfer interrupted by progress**

#### **core.error 3**

This message indicates that the data-transfer is interrupted by a timed progress callback and Zorp is closing the data-transfer channels.

### 1.1.75. Error reading request

#### **finger.error 1**

This message is appears when zorp cannot read finger request

### 1.1.76. Request details

#### **finger.request 6**

This message is say about read finger request

### 1.1.77. Parse error, dropping request

#### **finger.violation 1**

This message appear when zorp cannot parse request

### 1.1.78. Invalid character, dropping request

#### **finger.violation 1**

This message say that zorp found an invalid character in finger request.

### 1.1.79. Username too long, dropping request

#### **finger.violation 1**

This message is about too long username found in the request.

### 1.1.80. One hostname is too long in hostname chain, dropping request

#### **finger.violation 1**

This message is appear when a too long hostname found in the request hostname chain.

### 1.1.81. Error write request

#### **finger.error 1**

This message appear when some error found in server side.

### 1.1.82. Error write request

#### **finger.error 1**

This message appear when some error found in client side when writting the header.

### **1.1.83. Error write request**

#### **finger.error 1**

This message appear when some error found in client side when writting the response.

### **1.1.84. Error write request**

#### **finger.error 1**

This message appear when some error found in client side when writting the footer.

### **1.1.85. Policy violation, abort session**

#### **finger.policy 2**

This message is about administrator decision to reject the finger session.

### **1.1.86. Policy violation, drop session**

#### **finger.policy 2**

This message is about administrator decision to drop finger session.

### **1.1.87. fetching request**

#### **finger.debug 6**

This debug message is about proxy state when start to fetching request

### **1.1.88. asking policy**

#### **finger.debug 6**

This debug message is about proxy state when finger fetched request and asking policy about it

### **1.1.89. connecting server**

#### **finger.debug 6**

This debug message is about proxy state when finger start connect to server.

### **1.1.90. sending request**

#### **finger.debug 6**

This debug message is about proxy state when finger start send the request to server.

### **1.1.91. copying response**

**finger.debug 6**

This debug message is about proxy state when finger start to copy server answer to client.

### **1.1.92. everything is done**

**finger.debug 6**

This debug message is about proxy state when finger stop it's work.

### **1.1.93. Data connection accepted on client side**

**ftp.session 5**

This message reports that the data connection is accepted from the client.

### **1.1.94. Data connection accepted on server side**

**ftp.session 5**

This message reports that the data connection is accepted from the server.

### **1.1.95. Data connection established on client side**

**ftp.session 5**

This message reports that the data connection is established to the client.

### **1.1.96. Connected to client, but connection is not expected**

**ftp.error 4**

This message indicates that the data connection is established to the client, but the no connection is expected or the connection is canceled meanwhile.

### **1.1.97. Data connection established on server side**

**ftp.session 5**

This message reports that the data connection is established to the server.

### **1.1.98. Connected to server, but connection is not expected**

**ftp.error 4**

This message indicates that the data connection is established to the server, but the no connection is expected or the connection is canceled meanwhile.





### **1.1.99. Internal error, previous dispatcher not unregistered**

#### **ftp.error 3**

This message indicates that the previous data connection was not completely teared down and a new one is about to accept. This message indicates an internal error, please contact the Balasys QA team (devel@balasys.hu).

### **1.1.100. Internal error, previous attach not unregistered**

#### **ftp.error 3**

This message indicates that the previous data connection was not completely teared down while a new connection is being established. This message indicates an internal error, please contact the Balasys QA team (devel@balasys.hu).

### **1.1.101. Internal error, previous attach not unregistered**

#### **ftp.error 3**

This message indicates that the previous data connection was not completely teared down and a new one is being established. This message indicates an internal error, please contact the Balasys QA team (devel@balasys.hu).

### **1.1.102. Internal error, previous dispatcher not unregistered**

#### **ftp.error 3**

This message indicates that the previous data connection was not completely teared down and a new one is about to accept. This message indicates an internal error, please contact the Balasys QA team (devel@balasys.hu).

### **1.1.103. Resetting data connection**

#### **ftp.debug 6**

This message indicates that the data connection is going to be destroyed.

### **1.1.104. Internal error, previous data connection is not closed properly**

#### **ftp.error 3**

This message indicates an internal error that a previous data connection was not completed successfully. This condition should not occur, though the event is harmless, an explicit data reset clears this state. Please contact the Balasys QA team (devel@balasys.hu).

### **1.1.105. Internal error, cannot start data proxy because peers are not yet connected**

#### **ftp.error 2**



This message indicates an internal error that the proxy is unable to start a stacked data proxy because either the data connection to the FTP client or the FTP server is not yet established. Please contact the Balasys QA team (devel@balasys.hu).

### **1.1.106. Internal error, current command descriptor does not specify data transfer**

#### **ftp.error 2**

This message indicates an internal error that the current command descriptor changed while the data connection was being set up, thus we are unable to know which direction is permitted during data transfer. Please contact the Balasys QA team (devel@balasys.hu).

### **1.1.107. Data connection established**

#### **ftp.session 4**

This message reports that data connection is established between the client and the server, and proxy is being stacked for the data transfer.

### **1.1.108. Possible bounce attack**

#### **ftp.policy 3**

This message indicates that the IP address of the data connection to be established differs from the IP of the control connection. This might be caused by a real bounce attack on FTP, or a erroneously configured NAT translation on the client or server side.

### **1.1.109. Possible bounce attack**

#### **ftp.policy 3**

This message indicates that the IP address of the data connection to be established differs from the IP of the control connection. This might be caused by a real bounce attack on FTP, or a erroneously configured NAT translation on the client or server side.

### **1.1.110. Possible bounce attack**

#### **ftp.policy 3**

This message indicates that the IP address of the data connection to be established differs from the IP of the control connection. This might be caused by a real bounce attack on FTP, or a erroneously configured NAT translation on the client or server side.

### **1.1.111. Possible bounce attack**

#### **ftp.policy 3**



This message indicates that the IP address of the data connection to be established differs from the IP of the control connection. This might be caused by a real bounce attack on FTP, or a erroneously configured NAT translation on the client or server side.

#### **1.1.112. Internal error, client side not connected**

##### **ftp.error 1**

This message indicates an internal error, please contact the Balasys QA team ([devel@balasys.hu](mailto:devel@balasys.hu)).

#### **1.1.113. Internal error, server side not connected**

##### **ftp.error 1**

This message indicates an internal error, please contact the Balasys QA team ([devel@balasys.hu](mailto:devel@balasys.hu)).

#### **1.1.114. Max\_line\_length above upper limit**

##### **ftp.policy 2**

This message indicates that the configured max\_line\_length is above upper limit and Zorp sets it to the upper limit.

#### **1.1.115. Max\_username\_length above max\_line\_length**

##### **ftp.policy 2**

This message indicates that the configured max\_username\_length is above max\_line\_length which does not make sense and Zorp sets it to the max\_line\_length.

#### **1.1.116. Max\_password\_length above max\_line\_length**

##### **ftp.policy 2**

This message indicates that the configured max\_password\_length is above max\_line\_length which does not make sense and Zorp sets it to the max\_line\_length.

#### **1.1.117. Max\_hostname\_length above max\_line\_length**

##### **ftp.policy 2**

This message indicates that the configured max\_hostname\_length is above max\_line\_length which does not make sense and Zorp sets it to the max\_line\_length.

#### **1.1.118. Error parsing valid\_chars\_username**

##### **ftp.policy 2**

This message indicates that the character set specified in the valid\_chars\_username attribute has a syntax error.



### **1.1.119. Internal error in stream write, side is wrong**

#### **ftp.error 1**

This message indicates an internal error, please contact the Balasys QA team ([devel@balasys.hu](mailto:devel@balasys.hu)).

### **1.1.120. Cannot write full line**

#### **ftp.error 4**

This message reports that Zorp was unable to write out a full line and some data remained in the buffer.

### **1.1.121. Server answer doesn't begin with number**

#### **ftp.violation 1**

This message indicates that the server's answer does not begin with a valid 3 character long number.

### **1.1.122. Response arrived**

#### **ftp.response 6**

This message reports that a valid answer is read from the server.

### **1.1.123. Error reading from server**

#### **ftp.error 1**

This message indicates that Zorp was unable to fetch the answer from the server. It is likely caused by some timeout.

### **1.1.124. Line is too short to be a valid answer**

#### **ftp.violation 1**

This message indicates that too short answer is read from the server. A valid answer must be at least 4 character long.

### **1.1.125. Server answer has wrong continuation mark**

#### **ftp.violation 1**

This message indicates that the server's answer has invalid continuation mark.

### **1.1.126. Rejected answer**

#### **ftp.policy 3**

This message indicates that the given response is rejected and changed by the policy.

### **1.1.127. Error reading from client**

#### **ftp.error 2**

This message indicates that Zorp was unable to read from the client side.

### **1.1.128. Request fetched**

#### **ftp.request 6**

This message reports that a valid request is fetched from the client.

### **1.1.129. Empty command. Aborting**

#### **ftp.violation 1**

This message indicates that an empty command is received, and policy does not permit it.

### **1.1.130. Unknown command. Aborting**

#### **ftp.violation 1**

This message indicates that an unknown command is received and policy does not permit it.

### **1.1.131. Internal error, known command but command parse is unset**

#### **ftp.error 1**

This message indicates an internal error, please contact the Balasys QA team ([devel@balasys.hu](mailto:devel@balasys.hu)).

### **1.1.132. This command not allowed in non-transparent mode**

#### **ftp.error 3**

This message indicates that the given request was not permitted in non-transparent mode before the sever connection was established. It is likely caused by an AUTH command.

### **1.1.133. Request rejected**

#### **ftp.policy 3**

This message indicates that the given request was rejected by the policy.

### **1.1.134. Proxy answer**

#### **ftp.policy 4**

This message reports that the given request is answered by Zorp without sending the request to the server.



### 1.1.135. Rejected command (aborting)

**ftp.policy 2**

This message indicates that the given request is rejected and the connection is aborted by Zorp.

### 1.1.136. Bad policy type, aborting

**ftp.policy 1**

This message indicates an internal error, please contact the Balasys QA team (devel@balasys.hu).

### 1.1.137. Too many continuous lines in the answer

**ftp.policy 3**

\* This message reports that the server send an answer \* which have too many lines. Increase the self.max\_continuous\_line variable \* if this is not a security incident.

### 1.1.138. Reading from peer

**ftp.debug 8**

This message reports that Zorp is reading from it's peers on the given side.

### 1.1.139. Invalid port specified in non-transparent destination

**ftp.policy 3**

This message indicates that the port part of the username in non-transparent mode is not permitted by the policy and Zorp rejects the request. Check the 'target\_port\_range' attribute.

### 1.1.140. Invalid character in username

**ftp.policy 3**

This message indicates that the username sent by the client contains invalid characters and Zorp rejects the request. Check the 'valid\_chars\_username' attribute.

### 1.1.141. Hostname specified in username is too long

**ftp.policy 3**

This message indicates that the hostname part of the username in non-transparent mode is too long and Zorp rejects the request. Check the 'max\_hostname\_length' attribute.

### 1.1.142. Username too long

**ftp.policy 3**



This message indicates that the username is too long and Zorp rejects the request. Check the 'max\_username\_length' attribute.

### **1.1.143. Username too long**

#### **ftp.policy 3**

This message indicates that the username is too long and Zorp rejects the request. Check the 'max\_username\_length' attribute.

### **1.1.144. Invalid character in username**

#### **ftp.policy 3**

This message indicates that the username sent by the client contains invalid characters and Zorp rejects the request. Check the 'valid\_chars\_username' attribute.

### **1.1.145. Internal error, proxy in unknown state**

#### **ftp.error 1**

This message indicates an internal error, please contact the Balasys QA team (devel@balasys.hu).

### **1.1.146. Internal error, proxy in unknown state**

#### **ftp.error 1**

This message indicates an internal error, please contact the Balasys QA team (devel@balasys.hu).

### **1.1.147. Password too long**

#### **ftp.policy 3**

This message indicates that the password is too long and Zorp rejects the request. Check the 'max\_password\_length' attribute.

### **1.1.148. Password too long**

#### **ftp.policy 3**

This message indicates that the password is too long and Zorp rejects the request. Check the 'max\_password\_length' attribute.

### **1.1.149. Internal error, proxy in unknown state**

#### **ftp.error 1**

This message indicates an internal error, please contact the Balasys QA team (devel@balasys.hu).



### **1.1.150. Internal error, proxy in unknown state**

#### **ftp.error 1**

This message indicates an internal error, please contact the Balasys QA team ([devel@balasys.hu](mailto:devel@balasys.hu)).

### **1.1.151. Internal error, proxy in unknown state**

#### **ftp.error 1**

This message indicates an internal error, please contact the Balasys QA team ([devel@balasys.hu](mailto:devel@balasys.hu)).

### **1.1.152. Unexpected response to data transfer command**

#### **ftp.violation 1**

This message indicates that the data transfer command's answer sent by the server is invalid and Zorp resets the data transfer.

### **1.1.153. Internal error, proxy in unknown state**

#### **ftp.error 1**

This message indicates an internal error, please contact the Balasys QA team ([devel@balasys.hu](mailto:devel@balasys.hu)).

### **1.1.154. Invalid parameter for command**

#### **ftp.violation 3**

This message indicates that the parameter of the request is invalid and Zorp rejects the request. This request must not have any parameter at all.

### **1.1.155. Invalid parameter for command**

#### **ftp.violation 3**

This message indicates that the parameter of the request is invalid and Zorp rejects the request. This request must not have any parameter at all.

### **1.1.156. Internal error, proxy in unknown state**

#### **ftp.error 1**

This message indicates an internal error, please contact the Balasys QA team ([devel@balasys.hu](mailto:devel@balasys.hu)).

### **1.1.157. Missing parameter for the TYPE command**

#### **ftp.violation 2**





This message indicates that the required parameter for the TYPE command is missing and Zorp rejects the request.

### **1.1.158. Valid, but unsupported transfer type specification**

#### **ftp.error 3**

This message indicates that the requested transfer type specification is normally valid but currently unsupported by the proxy and Zorp rejects the request.

### **1.1.159. Unknown transfer type specification**

#### **ftp.violation 2**

This message indicates that the requested transfer type specification is invalid and Zorp rejects the request.

### **1.1.160. Missing parameter to the MODE command**

#### **ftp.violation 2**

This message indicates that the required parameter for the MODE command is missing and Zorp rejects the request.

### **1.1.161. Invalid parameter to the MODE command**

#### **ftp.violation 2**

This message indicates that the MODE command parameter is invalid and Zorp rejects the request.

### **1.1.162. Missing parameter to the STRU command**

#### **ftp.violation 2**

This message indicates that the required parameter for the STRU command is missing and Zorp rejects the request.

### **1.1.163. Invalid parameter to the STRU command**

#### **ftp.violation 2**

This message indicates that the STRU command parameter is invalid and Zorp rejects the request.

### **1.1.164. Error preparing server-side data connection (PORT)**

#### **ftp.error 2**

This message indicates that Zorp was unable to start listening for the data connection on the server side and Zorp rejects the request.



### **1.1.165. There was an error binding a server-side listener**

#### **ftp.error 2**

This message indicates that Zorp was unable to start listening for the data connection on the server side and Zorp rejects the request.

### **1.1.166. Error parsing PASV response**

#### **ftp.violation 2**

This message indicates that the response to the PASV command is invalid and Zorp rejects the response.

### **1.1.167. Error preparing data connection to the server (PASV)**

#### **ftp.error 2**

This message indicates that the proxy was unable to connect to the server on the port specified in its PASV response and Zorp rejects the response.

### **1.1.168. Error preparing server-side data connection listener (EPRT)**

#### **ftp.error 2**

This message indicates that Zorp was unable to start listening for the data connection on the server side and Zorp rejects the request.

### **1.1.169. Cannot bind to the given address (EPRT)**

#### **ftp.error 2**

This message indicates that Zorp was unable to start listening for the data connection on the server side and Zorp rejects the request.

### **1.1.170. Missing parameter (EPSV)**

#### **ftp.violation 2**

This message indicates that the required parameter for the EPSV command is missing and Zorp rejects the response.

### **1.1.171. Bad parameter (EPSV), not beginning with bracket**

#### **ftp.violation 2**

This message indicates that the parameter of the EPSV response does not begin with a bracket and Zorp rejects the response.



### **1.1.172. Bad parameter (EPSV), not closing with bracket**

#### **ftp.violation 2**

This message indicates that the parameter of the EPSV response does not have a closing bracket and Zorp rejects the response.

### **1.1.173. Error parsing EPSV response**

#### **ftp.violation 2**

This message indicates that the EPSV command response is invalid and Zorp rejects the response.

### **1.1.174. Bad parameter (EPSV), invalid port**

#### **ftp.violation 2**

This message indicates that the port number of the EPSV command response is invalid and Zorp rejects the response.

### **1.1.175. Internal error, cannot detect server address**

#### **ftp.violation 2**

This message indicates that Proxy cannot detect the server address. This an internal error.

### **1.1.176. Unknown protocol type (EPSV)**

#### **ftp.violation 1**

This message indicates that the protocol specified by the EPSV command response is not supported by the proxy and Zorp rejects the response.

### **1.1.177. Error preparing data connection to the server (EPSV)**

#### **ftp.error 2**

This message indicates that the proxy was unable to connect to the server on the port specified in its EPSV response and Zorp rejects the response.

### **1.1.178. Invalid parameters to the PORT command**

#### **ftp.violation 2**

This message indicates that the parameter of the PORT command is invalid and Zorp rejects the request.

### **1.1.179. Connection mode not supported**

#### **ftp.policy 1**



This message indicates that the 'data\_mode' attribute of the policy is invalid and Zorp rejects the request. Check the 'data\_mode' attribute.

### **1.1.180. Error preparing client-side data connection (PORT-<;PASV)**

#### **ftp.error 2**

This message indicates that the proxy was unable to connect to the client on the port specified in its PORT response and Zorp rejects the response.

### **1.1.181. Error parsing the server answer to the PASV command (PORT-<;PASV)**

#### **ftp.violation 2**

This message indicates that the response of the PASV command is invalid and Zorp rejects the response.

### **1.1.182. Error preparing client-side data connection (PORT)**

#### **ftp.error 2**

This message indicates that the proxy was unable to connect to the client on the port specified in its PORT response and Zorp rejects the response.

### **1.1.183. Error parsing the server answer to the PORT command**

#### **ftp.violation 2**

This message indicates that the response of the PORT command is invalid and Zorp rejects the response.

### **1.1.184. Connection mode not supported**

#### **ftp.policy 1**

This message indicates that the 'data\_mode' attribute of the policy is invalid and Zorp rejects the request. Check the 'data\_mode' attribute.

### **1.1.185. Error preparing client-side data connection listener (PASV)**

#### **ftp.error 2**

This message indicates that Zorp was unable to start listening for the data connection on the client side and Zorp rejects the request.

### **1.1.186. Error preparing client-side data connection listener (PASV)**

#### **ftp.error 2**

This message indicates that Zorp was unable to start listening for the data connection on the client side and Zorp rejects the request.



### **1.1.187. Error preparing client-side data connection listener (PASV-<;PORT)**

#### **ftp.error 2**

This message indicates that Zorp was unable to start listening for the data connection on the client side and Zorp rejects the request.

### **1.1.188. Error preparing client-side data connection listener (PASV-<;PORT)**

#### **ftp.error 2**

This message indicates that Zorp was unable to start listening for the data connection on the client side and Zorp rejects the request.

### **1.1.189. Error parsing the server answer to the PORT command (PASV-<;PORT)**

#### **ftp.violation 2**

This message indicates that the response of the PORT command is invalid and Zorp rejects the response.

### **1.1.190. Missing parameter (EPRT)**

#### **ftp.violation 2**

This message indicates that the required parameter for the EPRT command is missing and Zorp rejects the request.

### **1.1.191. Bad parameter (EPRT)**

#### **ftp.violation 2**

This message indicates that the parameter of the EPRT command is invalid and Zorp rejects the request.

### **1.1.192. Bad port parameter (EPRT)**

#### **ftp.violation 2**

This message indicates that the port number of the EPRT command is invalid and Zorp rejects the request.

### **1.1.193. Unknown protocol method (EPRT)**

#### **ftp.violation 2**

This message indicates that the protocol specified by the EPRT command is not supported by the proxy and Zorp rejects the response.

### **1.1.194. Bad host address (EPRT)**

#### **ftp.violation 2**



This message indicates that the host address of the EPRT command is invalid and Zorp rejects the request.

### **1.1.195. Connection mode not supported**

#### **ftp.policy 1**

This message indicates that the 'data\_mode' attribute of the policy is invalid and Zorp rejects the request. Check the 'data\_mode' attribute.

### **1.1.196. Error preparing client connect (EPRT)**

#### **ftp.error 2**

This message indicates that the proxy was unable to connect to the client specified in the EPRT response and Zorp rejects the response.

### **1.1.197. Bad server answer (EPRT)**

#### **ftp.violation 2**

This message indicates that the response of the EPRT is invalid and Zorp rejects the response.

### **1.1.198. Error preparing client connect (EPRT)**

#### **ftp.error 2**

This message indicates that the proxy was unable to connect to the client specified in the EPRT response and Zorp rejects the response.

### **1.1.199. Connection mode not supported**

#### **ftp.policy 1**

This message indicates that the 'data\_mode' attribute of the policy is invalid and Zorp rejects the request. Check the 'data\_mode' attribute.

### **1.1.200. Error preparing client listen (EPSV)**

#### **ftp.error 2**

This message indicates that Zorp was unable to start listening for the data connection on the client side and Zorp rejects the response.

### **1.1.201. Error preparing client listen (EPSV)**

#### **ftp.error 2**

This message indicates that Zorp was unable to start listening for the data connection on the client side and Zorp rejects the response.



### 1.1.202. Error preparing client listen (EPSV)

#### ftp.error 2

This message indicates that Zorp was unable to start listening for the data connection on the client side and Zorp rejects the response.

### 1.1.203. Error preparing client listen (EPSV)

#### ftp.error 2

This message indicates that Zorp was unable to start listening for the data connection on the client side and Zorp rejects the response.

### 1.1.204. Bad server answer (EPSV)

#### ftp.violation 2

This message indicates that the response of the EPSV command is invalid and Zorp rejects the response.

### 1.1.205. Error parsing command (ALLO)

#### ftp.violation 2

This message indicates that the parameter of the ALLO command is invalid and Zorp rejects the request.

### 1.1.206. Invalid stacking tuple returned by policy

#### ftp.policy 3

This message indicates that the request\_stack or response\_stack hash contains an invalid stacking tuple. It should contain a (stack\_type, proxy\_class) tuple. Check your Zorp configuration.

### 1.1.207. Invalid stacking type

#### ftp.policy 3

This message indicates that the request\_stack or response\_stack hash contains an invalid stacking type. Check your Zorp configuration.

### 1.1.208. Invalid request, data transfer failed

#### ftp.error 2

This message indicates that the processed request was invalid, and the data transfer failed.

### 1.1.209. Data transfer failed

#### ftp.error 2

This message reports that the data transfer failed.

### **1.1.210. Stacked proxy decision**

#### **ftp.error 2**

This message indicates that the stacked proxy returned the specified verdict about the content. Check the stacked proxy log for further information.

### **1.1.211. Stacked proxy accepted data**

#### **ftp.debug 6**

This message indicates that the stacked proxy accepted the content.

### **1.1.212. Policy does not contain this request, using hard-coded default**

#### **ftp.policy 5**

This message indicates that the policy does not contain any setting for the given request and Zorp rejects the request. Check the 'request' attribute.

### **1.1.213. Policy type invalid**

#### **ftp.policy 1**

This message indicates that the policy type is invalid for the given request and Zorp rejects the request. Check the 'request' attribute.

### **1.1.214. Cannot parse policy line**

#### **ftp.policy 1**

This message indicates that the policy for the given request is invalid and Zorp rejects the request. Check the 'request' attribute. It is likely that the parameter for the FTP\_REQ\_POLICY is invalid.

### **1.1.215. Error in policy calling**

#### **ftp.policy 1**

This message indicates that the callback for the given request policy is invalid and Zorp rejects the request. Check the 'request' attribute. It is likely that the parameter for the FTP\_REQ\_POLICY is invalid.

### **1.1.216. Can't parsing return code**

#### **ftp.policy 1**

This message indicates that the returned value of the callback for the given request policy is invalid and Zorp rejects the request. Check the callback function.





### **1.1.217. Policy does not contain this response, using hard-coded default**

#### **ftp.policy 5**

This message indicates that the policy does not contain any setting for the given response and Zorp rejects the response. Check the 'response' attribute.

### **1.1.218. Answer type invalid**

#### **ftp.policy 1**

This message indicates that the policy type is invalid for the given response and Zorp rejects the request. Check the 'request' attribute.

### **1.1.219. Bad policy line**

#### **ftp.policy 1**

This message indicates that the policy for the given response is invalid and Zorp rejects the response. Check the 'response' attribute. It is likely that the parameter for the FTP\_RSP\_POLICY is invalid.

### **1.1.220. Error in policy calling**

#### **ftp.policy 1**

This message indicates that the callback for the given response policy is invalid and Zorp rejects the response. Check the 'response' attribute. It is likely that the parameter for the FTP\_RSP\_POLICY is invalid.

### **1.1.221. Return code invalid from policy function**

#### **ftp.policy 1**

This message indicates that the returned value of the callback for the given response policy is invalid and Zorp rejects the response. Check the callback function.

### **1.1.222. Policy does not contain this feature, dropping**

#### **ftp.policy 5**

This message indicates that the policy does not contain any setting for the given feature and Zorp drops the feature. Check the 'features' attribute.

### **1.1.223. Policy value invalid**

#### **ftp.policy 1**

This message indicates that the policy type is invalid for the given feature and thus Zorp drops the feature.

### **1.1.224. Internal error, error code out of range**

#### **http.error 2**

This message indicates that Zorp caught an invalid error code internally. Please report this event to the BalaSys Development Team (at [devel@balasys.hu](mailto:devel@balasys.hu)).

### **1.1.225. An error occurred, would serve error file, but silent mode is enabled**

#### **http.debug 6**

This message reports that Zorp would send the given error page to the browser, if silent mode would not be enabled. It is likely that some protocol/configuration/proxy error occurred.

### **1.1.226. An error occurred, serving error file**

#### **http.debug 6**

This message reports that Zorp is sending the given error page to the clients browser. It is likely that some protocol/configuration/proxy error occurred.

### **1.1.227. Error writing stream, stream is NULL**

#### **http.error 1**

This message reports that Zorp was about to write to an invalid stream. Please report this event to the BalaSys Development Team (at [devel@balasys.hu](mailto:devel@balasys.hu)).

### **1.1.228. Error writing stream**

#### **http.error 1**

This message reports that Zorp was unable to write to the given stream. It is likely that the peer closed the connection unexpectedly.

### **1.1.229. Only Basic authentication is supported**

#### **http.error 3**

This message indicates that the client tried to use the given unsupported HTTP authentication. Currently only Basic HTTP authentication is supported by Zorp.

### **1.1.230. Invalid base64 encoded username:password pair**

#### **http.violation 1**

This message indicates that the client sent a malformed username:password field, during the authentication phase.



### **1.1.231. No colon is found in the decoded username:password pair**

**http.violation 2**

This message indicates that the username:password field received during authentication was malformed.

### **1.1.232. Invalid HTTP request received**

**http.violation 2**

This message indicates that the client sent an invalid HTTP request to the proxy.

### **1.1.233. Found category match for URL**

**http.accounting 4**

This message indicates that the requested URL matched at

### **1.1.234. CONNECT method without version specification**

**http.violation 1**

This message indicates that the client sent a CONNECT method request, but it is only supported for HTTP/1.0 or later.

### **1.1.235. This request type is not permitted in transparent mode**

**http.policy 2**

This message indicates that the client sent the given type request, which is not permitted by the policy. Check the permit\_proxy\_requests and the permit\_server\_requests attributes.

### **1.1.236. No host header in transparent request, 'unknown' is used instead**

**http.violation 4**

This message indicates that no Host header was sent by the client. As the content of the host header is used to reconstruct the requested URL, the request\_url attribute will refer to a host named 'unknown'.

### **1.1.237. 'Host:' header is required, and HTTP/0.9 can't transfer headers**

**http.policy 2**

This message indicates that an HTTP/0.9 request was sent by the client, and Host header is required by the policy, but HTTP/0.9 does not support headers. Check the require\_host\_header attribute.

### **1.1.238. No 'Host:' header in request, and policy requires this**

**http.policy 2**

This message indicates that no Host header was sent by the client, but it was required by the policy. Check the `require_host_header` attribute.

### **1.1.239. Error parsing URL**

#### **http.error 1**

This message indicates that there was an error parsing an already canonicalized URL. Please report this event to the BalaSys Development Team (at [devel@balasys.hu](mailto:devel@balasys.hu))

### **1.1.240. Error parsing URL**

#### **http.error 1**

This message indicates that there was an error parsing an already canonicalized URL. Please report this event to the BalaSys Development Team (at [devel@balasys.hu](mailto:devel@balasys.hu))

### **1.1.241. Client attempted to use FTP over HTTP, which is currently disabled**

#### **http.policy 2**

This message indicates that a client tried to use FTP over HTTP which is not allowed by default. Either set a parent proxy or enable the `permit_ftp_over_http` attribute.

### **1.1.242. Unsupported scheme in URL**

#### **http.error 3**

This message indicates that the requested URL refers to an unsupported protocol scheme. Zorp currently knows about `http` and `cache_object` protocols, and can support the `ftp` protocol if a parent proxy supporting ftp over http tunneling is present.

### **1.1.243. Too long hostname in URL**

#### **http.policy 2**

This message indicates that the HTTP request was rejected because the hostname part in the URL was too long. You can increase the permitted limit by changing the `max_hostname_length` attribute.

### **1.1.244. Accounting**

#### **http.accounting 4**

This is an accounting message that reports the requested method and URL.

### **1.1.245. Invalid item in request hash**

#### **http.policy 1**



This message indicates that the request hash contains an invalid item for the given request method. Check your Zorp configuration.

#### **1.1.246. Error parsing HTTP\_REQ\_POLICY tuple in request hash**

##### **http.policy 1**

This message indicates that the request hash contains an invalid POLICY tuple for the given request method. It should contain a valid call-back function in the tuple.

#### **1.1.247. Error parsing HTTP\_REQ\_REJECT in request hash**

##### **http.policy 1**

This message indicates that the request hash contains an invalid REJECT tuple for the given request method. It should contain an error message, which is sent back to the client.

#### **1.1.248. Unknown request hash item**

##### **http.policy 1**

This message indicates that the request hash contains an invalid action for the given request method. Check your Zorp configuration.

#### **1.1.249. Policy requested to send custom response**

##### **http.policy 6**

This log message indicates that the policy requested a custom response to be sent to the client.

#### **1.1.250. Request not permitted by policy**

##### **http.policy 2**

This log message indicates that the specified HTTP request was not permitted by your policy.

#### **1.1.251. Request not permitted by policy**

##### **http.policy 2**

This log message indicates that the specified HTTP request was not permitted by your policy.

#### **1.1.252. Connecting to this port is prohibited by policy**

##### **http.violation 2**

This message indicates that the proxy did not allow addressing the specified port as the target\_port\_range attribute does not allow it.



### **1.1.253. Internal error initializing server stream**

#### **http.error 1**

This message indicates that initializing the server stream failed. Please report this event to the BalaSys Development Team (at [devel@balasys.hu](mailto:devel@balasys.hu)).

### **1.1.254. Error in HTTP/0.9 compatibility code, line buffer full**

#### **http.error 2**

This message indicates that Zorp was unable to enable HTTP/0.9 compatibility mode, due to the full buffer. If you experience this problem many times, please contact your Zorp support.

### **1.1.255. Server falled back to HTTP/0.9 which is prohibited by policy**

#### **http.policy 2**

This message indicates that the server sent back HTTP/0.9 response, which is prohibited by the policy. It is likely a buggy or old server. Check the `permit_http09_responses` attribute.

### **1.1.256. Invalid HTTP response heading**

#### **http.violation 1**

This message indicates the the HTTP status line returned by the server was invalid.

### **1.1.257. Error parsing response version**

#### **http.violation 1**

This message indicates that the server sent the response with an invalid HTTP version. It is likely that the server is buggy.

### **1.1.258. Unsupported protocol version**

#### **http.violation 1**

This message indicates that the server sent an unsupported protocol version. It is likely that the server is buggy.

### **1.1.259. Invalid response hash item**

#### **http.policy 1**

This message indicates that the response hash contains an invalid item for the given response. Check your Zorp configuration.



### **1.1.260. Error parsing HTTP\_RSP\_POLICY in response hash**

#### **http.policy 1**

This message indicates that the response hash contains an invalid POLICY tuple for the given response. It should contain a valid call-back function in the tuple.

### **1.1.261. Error parsing HTTP\_RSP\_REJECT in response hash**

#### **http.policy 1**

This message indicates that the response hash contains an invalid REJECT tuple for the given response. It should contain an error message, which is sent back to the client.

### **1.1.262. Invalid response hash item**

#### **http.policy 1**

This message indicates that the response hash contains an invalid action for the given response. Check your Zorp configuration.

### **1.1.263. Response not permitted by policy**

#### **http.policy 2**

This message indicates that the status code returned by the server is not a permitted response code for this request.

### **1.1.264. Missing port number in CONNECT request**

#### **http.violation 1**

This message indicates that the received CONNECT request did not include a port number to connect to.

### **1.1.265. Parent proxy refused our CONNECT request**

#### **http.error 1**

This message indicates that the our parent proxy refused our CONNECT request. It is likely that the parent proxy does not permit CONNECT method.

### **1.1.266. Starting connect method**

#### **http.debug 6**

This message reports that CONNECT method is in use, and CONNECT method was accepted by our parent proxy.



### **1.1.267. Internal error, connectMethod is expected to return a proxy instance**

#### **http.error 1**

This message indicates that an internal error occurred, the connectMethod function did not return an integer. Please report this event to the BalaSys Development Team (at [devel@balasys.hu](mailto:devel@balasys.hu)).

### **1.1.268. Fetching request and headers**

#### **http.debug 6**

This message reports that Zorp is fetching the request and the headers from the client.

### **1.1.269. processing request and headers**

#### **http.debug 6**

This message reports that Zorp is processing the fetched request and the headers.

### **1.1.270. Maximum keep-alive request reached, setting connection mode to close**

#### **http.policy 3**

This message reports that the maximum number of requests in a keep-alive loop is reached, and Zorp is closing after this request. Check the `max_keepalive_request` attribute.

### **1.1.271. Filtering request and headers**

#### **http.debug 6**

This message reports that Zorp is filtering the processed request and the headers.

### **1.1.272. Reprocessing filtered request**

#### **http.debug 6**

This message indicates that Zorp is rechecking the HTTP request after possible changes performed by the policy layer.

### **1.1.273. Sending request and headers, copying request data**

#### **http.debug 6**

This message reports that Zorp is sending the filtered request and headers, and copies the requests data to the server.

### **1.1.274. Fetching response and headers**

#### **http.debug 6**





This message reports that Zorp is fetching the response and headers from the server.

### **1.1.275. Processing response and headers**

**http.debug 6**

This message reports that Zorp is processing the fetched response and the headers.

### **1.1.276. Filtering response and headers**

**http.debug 6**

This message reports that Zorp is filtering the processed response and the headers.

### **1.1.277. Copying response and headers, copying response data**

**http.debug 6**

This message reports that Zorp is sending the filtered response and headers, and copies the response data to the client.

### **1.1.278. Internal error, invalid server\_protocol**

**core.error 1**

This message indicates an internal error in HTTP proxy. Please report this event to the BalaSys Development Team (at [devel@balasys.hu](mailto:devel@balasys.hu)).

### **1.1.279. exiting keep-alive loop**

**http.debug 6**

This message reports that Zorp is exiting the keep-alive loop and closing the connection.

### **1.1.280. Chunk length line too long**

**http.violation 1**

This message indicates that the chunk length line is too long. It is likely caused by a buggy client or server.

### **1.1.281. Invalid chunk length**

**http.violation 1**

This message indicates that the chunk length is invalid. It is likely caused by a buggy client or server.

### **1.1.282. Invalid chunk length**

**http.violation 1**

This message indicates that the chunk length is invalid. It is likely caused by a buggy client or server.

### **1.1.283. Chunk too large**

#### **http.policy 2**

This message indicates that the length of the chunk is larger than allowed or is a negative number. Check the 'max\_chunk\_length' attribute.

### **1.1.284. Unexpected EOF while dechunking stream**

#### **http.violation 1**

This message indicates that Zorp unexpectedly got EOF during chunk encoded data transfer. It is likely a caused by a buggy client or server.

### **1.1.285. Chunk footer is not an empty line**

#### **http.violation 1**

This message indicates that the chunk footer contains data. It is likely caused by a buggy client or server.

### **1.1.286. Stacked proxy sent a content-length hint, using it**

#### **http.debug 6**

This message reports that the stacked proxy sent a content-length hint on how much data will be sent and the http proxy is using it to set the Content-Length header.

### **1.1.287. Invalid stacking tuple returned by policy**

#### **http.policy 3**

This message indicates that the request\_stack or response\_stack hash contains an invalid stacking tuple. It should contain a (stack\_type, proxy\_class) tuple. Check your Zorp configuration.

### **1.1.288. Invalid stacking type**

#### **http.policy 3**

This message indicates that the request\_stack or response\_stack hash contains an invalid stacking type. Check your Zorp configuration.

### **1.1.289. The header 'Content-Length' was present, but is not a number**

#### **http.violation 1**

This message indicates that the Content-Length headers value is not a valid number. It is likely caused by a buggy client or server.

### **1.1.290. Invalid request, data transfer failed**

**http.error 2**

This message indicates that the processed request was invalid, and the data transfer failed.

### **1.1.291. Data transfer failed**

**http.error 2**

This message reports that the data transfer failed.

### **1.1.292. Stacked proxy decision**

**http.error 2**

This message indicates that the stacked proxy returned the specified verdict about the content. Check the stacked proxy log for further information.

### **1.1.293. Stacked proxy accepted data**

**http.debug 6**

This message indicates that the stacked proxy accepted the content.

### **1.1.294. Invalid FTP response, line too short**

**http.violation 2**

This message indicates that the response given by the FTP server was too short, not even the mandatory status code was included.

### **1.1.295. Invalid FTP response, response code not numeric**

**http.violation 2**

This message indicates that the FTP server gave an invalid response, the status code returned by the server was not numeric.

### **1.1.296. Invalid FTP response, continuation line contains different status code**

**http.violation 2**

This message indicates that the FTP server gave an invalid response as the status code changed from the one which was present on the first line.

### **1.1.297. FTP server returned an unhandled status code for the login request**

**http.error 4**



This message indicates that the FTP server returned an unknown status code in response to our login request.

### **1.1.298. Error logging in**

#### **http.error 4**

This message indicates that the FTP refused our authentication attempt.

### **1.1.299. Error in FTP greeting, expecting 220**

#### **http.error 3**

This message indicates that the FTP server greeted Zorp with a non-220 response code.

### **1.1.300. Error changing directory on server**

#### **http.error 4**

This message indicates that the directory specified in the URL was not accepted by the FTP server.

### **1.1.301. Error changing directory on server**

#### **http.error 4**

This message indicates that the directory specified in the URL was not accepted by the FTP server.

### **1.1.302. Error changing directory on server**

#### **http.error 4**

This message indicates that the directory specified in the URL was not accepted by the FTP server.

### **1.1.303. Error sending RETR or LIST command to FTP server**

#### **http.error 4**

This message indicates that an I/O error occurred while sending our RETR or LIST request.

### **1.1.304. Error establishing data connection to FTP server**

#### **http.error 4**

This message indicates that an I/O error occurred while trying to establish the data connection with the FTP server.

### **1.1.305. Error reading data channel confirmation from FTP server**

#### **http.error 4**



This message indicates that the FTP server did not react with a 150 response after the data connection was established.

### **1.1.306. Error reading 226 from FTP server**

#### **http.error 4**

This message indicates that the FTP server did not return with an appropriate status code at the end of the data transfer.

### **1.1.307. Invalid HTTP\_HDR\_CHANGE\_NAME rule in header processing**

#### **http.policy 1**

This message indicates that the HDR\_CHANGE\_NAME parameter is invalid, for the given header. Check your request\_headers and response\_headers hashes.

### **1.1.308. Invalid HTTP\_HDR\_CHANGE\_VALUE rule in header processing**

#### **http.policy 1**

This message indicates that the HDR\_CHANGE\_VALUE parameter is invalid, for the given header. Check your request\_headers and response\_headers hashes.

### **1.1.309. Invalid HTTP\_HDR\_CHANGE\_BOTH rule in header processing**

#### **http.policy 1**

This message indicates that the HDR\_CHANGE\_BOTH parameter is invalid, for the given header. Check your request\_headers and response\_headers hashes.

### **1.1.310. Invalid value in header action tuple**

#### **http.policy 1**

This message indicates that the action is invalid, for the given header. Check your request\_headers and response\_headers hashes.

### **1.1.311. Error reading from peer while fetching headers**

#### **http.error 3**

This message indicates that Zorp was unable to fetch headers from the server. Check the permit\_null\_response attribute.

### **1.1.312. First header starts with white space**

#### **http.violation 2**



This message indicates that Zorp fetched an invalid header from the server. It is likely caused by a buggy server.

### **1.1.313. Invalid HTTP header**

#### **http.violation 2**

This message indicates that the server sent an invalid HTTP header.

### **1.1.314. Too many header lines**

#### **http.policy 2**

This message indicates that the server tried to send more header lines, than the allowed maximum. Check the `max_header_lines` attribute.

### **1.1.315. Request method empty, or too long**

#### **http.violation 1**

This message indicates that the request method sent by the client is invalid.

### **1.1.316. URL missing, or too long**

#### **http.violation 1**

This message indicates that the URL sent by the client is invalid.

### **1.1.317. Protocol version missing, or too long**

#### **http.violation 1**

This message indicates that the protocol version sent by the client is invalid.

### **1.1.318. Request details**

#### **http.request 6**

This message reports the processed request details.

### **1.1.319. Invalid HTTP status line**

#### **http.response 6**

This message indicates that the server sent an invalid response status line.

### **1.1.320. Response version empty or too long**

#### **http.violation 1**



This message indicates that the protocol version sent by the server is invalid.

### **1.1.321. Response code empty or too long**

#### **http.violation 1**

This message indicates that the response code sent by the server is invalid.

### **1.1.322. Response details**

#### **http.response 7**

This message reports the processed response details.

### **1.1.323. Unknown protocol version**

#### **http.request 3**

This message indicates that the protocol version sent by the client is unsupported.

### **1.1.324. Unknown protocol version**

#### **http.response 3**

This message indicates that the protocol version sent by the server is unsupported.

### **1.1.325. Invalid return value of packetStats(), integer required**

#### **plug.policy 1**

This message is logged when the policy layer returned a non-integer value in its packetStats() function. packetStats() is expected to return ZV\_REJECT or ZV\_ACCEPT.

### **1.1.326. packetStats() requested to abort session**

#### **plug.policy 1**

This message indicates that the verdict returned by the packetStats() function requests to terminate the session.

### **1.1.327. Response line too long**

#### **pop3.violation 3**

This message indicates that the response line is too long and Zorp aborts the connection. Check the 'max\_reply\_length' attribute.

### **1.1.328. Response status is invalid**

#### **pop3.violation 3**



This message indicates that the status of the response is invalid and Zorp rejects the response. The response should begin with '+OK' or with '-ERR'.

### **1.1.329. Response fetched with parameter**

**pop3.response 7**

This message reports that the fetched response contains a parameter.

### **1.1.330. Response fetched**

**pop3.response 7**

This message reports the fetched response.

### **1.1.331. Data transfer failed**

**pop3.error 2**

This message indicates that the multi-line data transfer failed and Zorp rejects the response.

### **1.1.332. Request line too long**

**pop3.violation 3**

This message indicates that the request line is too long and Zorp rejects the request. Check the 'max\_request\_length' attribute.

### **1.1.333. Request fetched with parameter**

**pop3.request 7**

This message reports that the fetched request contains a parameter.

### **1.1.334. Request fetched**

**pop3.request 7**

This message reports the fetched request.

### **1.1.335. Unknown request command**

**pop3.request 3**

This message indicates that the request was unknown and Zorp aborts the connection. Check the 'permit\_unknown\_command' and the 'request' attributes.



### **1.1.336. Request command not allowed in this state**

**pop3.request 3**

This message indicates that the request command is not allowed in this state of the protocol and Zorp rejects the request.

### **1.1.337. Dropping request parameter, no parameter allowed**

**pop3.request 4**

This message indicates that the request must not have any parameter and Zorp is going to drop the parameter.

### **1.1.338. The required numerical parameter of the request is missing**

**pop3.request 3**

This message indicates that the numerical parameter of the request is missing and Zorp aborts the connection.

### **1.1.339. The numerical parameter of the request is not in the given range**

**pop3.request 3**

This message indicates that the numerical parameter of the request is not in the given range and Zorp aborts the connection.

### **1.1.340. The numerical parameter of the request is negative**

**pop3.request 3**

This message indicates that the numerical parameter of the request is a negative number which is invalid and Zorp aborts the connection.

### **1.1.341. The numerical parameter of the request is zero**

**pop3.request 3**

This message indicates that the numerical parameter of the request is zero which is invalid and Zorp aborts the connection.

### **1.1.342. The numerical parameter of the request contains junk after the number**

**pop3.request 4**

This message indicates that the numerical parameter of the request contains junk characters after the number but Zorp ignores and truncates the junk.



### **1.1.343. The first numerical parameter of the request is not in the given range**

**pop3.request 3**

This message indicates that the first numerical parameter is not in the given range and Zorp aborts the connection.

### **1.1.344. The first numerical parameter of the request is negative**

**pop3.request 3**

This message indicates that the first numerical parameter of the request is a negative number which is invalid and Zorp aborts the connection.

### **1.1.345. Only one numerical argument in request**

**pop3.request 3**

This message indicates that only one numerical parameter is present but two is required and Zorp rejects the request.

### **1.1.346. The second numerical parameter of the request is not in the given range**

**pop3.request 3**

This message indicates that the second numerical parameter is not in the given range and Zorp aborts the connection.

### **1.1.347. The second numerical parameter of the request is a negative number**

**pop3.request 3**

This message indicates that the second numerical parameter of the request is a negative number which is invalid and Zorp aborts the connection.

### **1.1.348. The numerical parameter of the request contain junk after the number**

**pop3.request 4**

This message indicates that the numerical parameters of the request contain junk characters after the numbers but Zorp ignores and truncates the junk.

### **1.1.349. Username is too long**

**pop3.policy 2**

This message indicates that the username parameter of the request is too long and Zorp rejects the request. Check the 'max\_username\_length' attribute.

### **1.1.350. Password is too long**

#### **pop3.policy 2**

This message indicates that the password parameter of the request is too long and Zorp rejects the request. Check the 'max\_password\_length' attribute.

### **1.1.351. The username parameter is too long or the digest parameter is missing**

#### **pop3.request 3**

This message indicates that the username parameter is too long or the digest missing after the username and Zorp rejects the request.

### **1.1.352. Error parsing the MD5 digest**

#### **pop3.request 3**

This message indicates that the MD5 digest parameter of the request is invalid and Zorp rejects the request.

### **1.1.353. The required numerical parameter of the response is missing**

#### **pop3.response 3**

This message indicates that the numerical parameter of the response is missing and Zorp aborts the connection.

### **1.1.354. The numerical parameter of the response is not in the given range**

#### **pop3.response 3**

This message indicates that the numerical parameter of the response is not in the given range and Zorp aborts the connection.

### **1.1.355. The numerical parameter of the response is a negative number**

#### **pop3.response 3**

This message indicates that the numerical parameter of the response is a negative number which is invalid and Zorp aborts the connection.

### **1.1.356. The numerical parameter of the response contains junk after the number**

#### **pop3.response 4**

This message indicates that the numerical parameter of the response contains junk characters after the number but Zorp ignores and truncates the junk.



### **1.1.357. The numerical parameter of the response is not in the given range**

**pop3.response 3**

This message indicates that the numerical parameter of the response is not in the given range and Zorp aborts the connection.

### **1.1.358. The required numerical parameter of the response is missing**

**pop3.response 3**

This message indicates that the numerical parameter of the response is missing and Zorp aborts the connection.

### **1.1.359. The numerical parameter of the response is a negative number**

**pop3.response 3**

This message indicates that the numerical parameter of the response is a negative number which is invalid and Zorp aborts the connection.

### **1.1.360. The second numerical parameter of the response is not in the given range**

**pop3.response 3**

This message indicates that the second numerical parameter of the response is not in the given range and Zorp aborts the connection.

### **1.1.361. The required second numerical parameter of the response is missing**

**pop3.response 3**

This message indicates that the second numerical parameter of the response is missing and Zorp aborts the connection.

### **1.1.362. The second numerical parameter of the response is a negative number**

**pop3.response 3**

This message indicates that the second numerical parameter of the response is a negative number which is invalid and Zorp aborts the connection.

### **1.1.363. The second numerical parameter of the response contains junk after the number**

**pop3.request 4**

This message indicates that the second numerical parameter of the response contains junk characters after the number but Zorp ignores and truncates the junk.

### **1.1.364. Stacked proxy rejected contents**

#### **pop3.error 2**

This message indicates that the stacked proxy rejected the content and Zorp rejects the response.

### **1.1.365. Policy does not contain this request, using hard-coded default**

#### **pop3.debug 5**

This message indicates that the policy does not contain any setting for the given request and Zorp rejects the request. Check the 'request' attribute.

### **1.1.366. Policy type is invalid**

#### **pop3.policy 1**

This message indicates that the policy type is invalid for the given request and Zorp aborts the connection. Check the 'request' attribute.

### **1.1.367. Cannot parse policy line**

#### **pop3.policy 1**

This message indicates that the policy for the given request is invalid and Zorp aborts the connection. Check the 'request' attribute. It is likely that the parameter for the POP3\_REQ\_POLICY is invalid.

### **1.1.368. Error in policy call**

#### **pop3.policy 1**

This message indicates that the callback for the given request policy is invalid and Zorp aborts the connection. Check the 'request' attribute. It is likely that the parameter for the POP3\_REQ\_POLICY is invalid.

### **1.1.369. Cannot parse the return code**

#### **pop3.policy 1**

This message indicates that the returned value of the callback for the given request policy is invalid and Zorp aborts the connection. Check the callback function.

### **1.1.370. Policy does not contain this request, using hard-coded default**

#### **pop3.debug 5**

This message indicates that the policy does not contain any setting for the given response and Zorp rejects the response. Check the 'request' attribute.



### **1.1.371. Policy type is invalid**

#### **pop3.policy 1**

This message indicates that the policy type is invalid for the given response and Zorp aborts the connection. Check the 'request' attribute.

### **1.1.372. Cannot parse policy line**

#### **pop3.policy 1**

This message indicates that the policy for the given request/response is invalid and Zorp aborts the connection. Check the 'request' attribute. It is likely that the parameter for the POP3\_REQ\_POLICY is invalid.

### **1.1.373. Error in policy call**

#### **pop3.policy 1**

This message indicates that the callback for the given request policy is invalid and Zorp aborts the connection. Check the 'request' attribute. It is likely that the parameter for the POP3\_REQ\_POLICY is invalid.

### **1.1.374. Cannot parse return code**

#### **pop3.policy 1**

This message indicates that the returned value of the callback for the given response policy is invalid and Zorp aborts the connection. Check the callback function.

### **1.1.375. Stack policy type is invalid**

#### **pop3.policy 1**

This message indicates that the stack policy type is invalid for the given response, so nothing will be stacked. Check the 'response\_stack' attribute.

### **1.1.376. Cannot parse stack policy line**

#### **pop3.policy 1**

This message indicates that the stack policy for the given response is invalid and Zorp stacks nothing. Check the 'response\_stack' attribute. It is likely that the parameter for the POP3\_STK\_MIME or POP3\_STK\_DATA is invalid.

### **1.1.377. Cannot parse stack policy line**

#### **pop3.policy 1**

This message indicates that the stack policy for the given response is invalid and Zorp stacks nothing. Check the 'response\_stack' attribute. It is likely that the parameter for the POP3\_STK\_POLICY is invalid.

### **1.1.378. Error in policy call**

#### **pop3.policy 1**

This message indicates that the callback for the given request policy is invalid and Zorp stacks nothing. Check the 'request' attribute. It is likely that the parameter for the POP3\_STK\_POLICY is invalid.

### **1.1.379. Cannot parse return code**

#### **pop3.policy 1**

This message indicates that the returned value of the callback for the given response policy is invalid and Zorp stacks nothing. Check the callback function.

### **1.1.380. Invalid command verb in request**

#### **smtp.violation 2**

This message indicates that the request command verb is invalid and Zorp rejects the request.

### **1.1.381. Request parsed**

#### **smtp.request 7**

This message reports that the request was parsed successfully.

### **1.1.382. Fetching request**

#### **smtp.debug 6**

This message reports that the request is going to be fetched.

### **1.1.383. Request line too long**

#### **smtp.violation 2**

This message indicates that the request line is too long and Zorp rejects the request. Check the 'max\_request\_length' attribute.

### **1.1.384. Processing request**

#### **smtp.debug 6**

This message reports that the request is going to be processed.

### **1.1.385. Command not permitted in this state**

#### **smtp.violation 4**

This message indicates that the given command is not permitted in this state of the communication and Zorp rejects the request.

### **1.1.386. Unknown command**

#### **smtp.violation 2**

This message indicates that the given request is unknown and Zorp rejects it. Check the 'permit\_unknown\_command' and the 'request' attributes.

### **1.1.387. Copying request to server**

#### **smtp.debug 6**

This message reports that the request is going to be copied to the server.

### **1.1.388. Error sending request**

#### **smtp.error 3**

This message indicates that an error occurred during sending the request to the server.

### **1.1.389. Fetching authentication request**

#### **smtp.debug 6**

This message reports that the authentication request is going to be fetched.

### **1.1.390. Auth request line too long**

#### **smtp.violation 2**

This message indicates that the authentication request line is too long and Zorp rejects the request. Check the 'max\_auth\_request\_length' attribute.

### **1.1.391. Copying authentication request to server**

#### **smtp.debug 6**

This message reports that the authentication request is going to be copied to the server.

### **1.1.392. Error sending authentication request**

#### **smtp.error 3**

This message indicates that an error occurred during sending the authentication request to the server.





### **1.1.393. SMTP reply contains non-numeric characters**

**smtp.violation 2**

This message indicates that the response contains non-numeric characters and Zorp rejects the response.

### **1.1.394. Invalid continuation character**

**smtp.violation 2**

This message indicates that the continuation character of the response contains an invalid character and Zorp rejects the response. The response must contain ' ' or '-' after the response code.

### **1.1.395. Fetching response**

**smtp.debug 6**

This message reports that the response is going to be fetched.

### **1.1.396. Response line too long**

**smtp.violation 2**

This message indicates that the response line is too long and Zorp rejects the response. Check the 'max\_response\_length' attribute.

### **1.1.397. Invalid SMTP reply, reply code changed**

**smtp.violation 2**

This message indicates that the reply code has changed in the continuation lines and Zorp rejects the response.

### **1.1.398. Response parsed**

**smtp.response 7**

This message reports that the response is parsed successfully.

### **1.1.399. Processing response**

**smtp.debug 6**

This message indicates that the response is going to be processed.

### **1.1.400. Copying response to client**

**smtp.debug 6**

This message reports that the response is going to be copied to the client.



### **1.1.401. Error sending SMTP reply**

**smtp.error 3**

This message indicates that an error occurred during sending the response to the client.

### **1.1.402. Invalid policy ignored, allowing 250 response to NOOP is required**

**smtp.policy 3**

This message indicates that the response code 250 for the NOOP request is required and Zorp ignores the invalid policy while generating NOOPs to the server.

### **1.1.403. Invalid policy ignored, allowing 354 response to DATA is required**

**smtp.policy 3**

This message indicates that the response code 354 for the DATA request is required and Zorp ignores the invalid policy during data transfer to the server.

### **1.1.404. Invalid contents**

**smtp.policy 3**

This message indicates that the content was declared invalid by the stacked proxy and Zorp rejects it.

### **1.1.405. Error occurred while scanning contents**

**smtp.policy 3**

This message indicates that an error occurred during stacked proxy handle the contents and Zorp send back a temporary error message.

### **1.1.406. Rejecting empty message**

**smtp.error 3**

This message indicates that an empty message is received from the stacked proxy and Zorp rejects it.

### **1.1.407. Error sending RSET to the server**

**smtp.error 3**

This message indicates that Zorp was unable to send RSET command to the server.

### **1.1.408. Exiting SMTP loop**

**smtp.debug 6**



This message reports that Zorp is exiting from the SMTP proxy loop and closing connections on both side.

#### **1.1.409. Error parsing local part**

##### **smtp.violation 2**

This message indicates that parsing the local part of the address failed.

#### **1.1.410. Local part does not end in '@'**

##### **smtp.violation 2**

This message indicates that the local part of the mail is invalid because it does not end with a '@' sign.

#### **1.1.411. Invalid domain name in path**

##### **smtp.violation 2**

This message indicates that the domain name of the mail is invalid.

#### **1.1.412. Path does not start with '>'**

##### **smtp.violation 2**

This message indicates that the address path is invalid because it does not start with a '>' sign.

#### **1.1.413. Invalid source route information**

##### **smtp.violation 2**

This message indicates that the source-root information of the address patch is invalid.

#### **1.1.414. Invalid address information**

##### **smtp.violation 2**

This message indicates that the address information is invalid.

#### **1.1.415. Path does not begin with '>,' but ends with '<,'**

##### **smtp.violation 2**

This message indicates that the address path is invalid because it does not start with a '<,' sign but ends with it.

#### **1.1.416. Path begins with '>,' but does not end with '<,'**

##### **smtp.violation 2**

This message indicates that the address path is invalid because it does not end with a '<,' sign.



### **1.1.417. Invalid SIZE extension in the MAIL command**

#### **smtp.violation 2**

This message indicates that the SIZE extension of the MAIL command is invalid because it must contain non-numeric characters. Zorp rejects the request.

### **1.1.418. Invalid BODY extension in the MAIL command**

#### **smtp.violation 2**

This message indicates that the BODY extension of the MAIL command is invalid because it must contain either '7BIT' or '8BITMIME'. Zorp rejects the request.

### **1.1.419. Invalid AUTH sender, not an xtext**

#### **smtp.violation 2**

This message indicates that the AUTH extension of the MAIL command is invalid because it must be xtext. Zorp rejects the request.

### **1.1.420. Invalid extension in the MAIL command**

#### **smtp.violation 2**

This message indicates that the given extension is invalid with the MAIL command and Zorp rejects the request.

### **1.1.421. Invalid request policy type**

#### **smtp.policy 1**

This message indicates that the policy type is invalid for the given request and Zorp aborts the connection. Check the 'request' attribute.

### **1.1.422. Error in request policy**

#### **smtp.policy 1**

This message indicates that the parameter of the request policy of the given request is invalid and Zorp aborts the connection. Check the 'request' attribute.

### **1.1.423. Error in request policy**

#### **smtp.policy 1**

This message indicates that the parameter of the request policy of the given request is invalid and Zorp aborts the connection. Check the 'request' attribute.

### **1.1.424. The verdict returned by the policy is not an int**

#### **smtp.policy 1**

This message indicates that the returned value of the callback for the given request policy is invalid and Zorp aborts the connection. Check the callback function.

### **1.1.425. Invalid response policy**

#### **smtp.policy 1**

This message indicates that the policy type is invalid for the given response and Zorp aborts the connection. Check the 'response' attribute.

### **1.1.426. Error in response policy**

#### **smtp.policy 1**

This message indicates that the parameter of the response policy of the given request is invalid and Zorp aborts the connection. Check the 'response' attribute.

### **1.1.427. Error in response policy**

#### **smtp.policy 1**

This message indicates that the parameter of the response policy of the given request is invalid and Zorp aborts the connection. Check the 'response' attribute.

### **1.1.428. The verdict returned by the policy is not an int**

#### **smtp.policy 1**

This message indicates that the returned value of the callback for the given response policy is invalid and Zorp aborts the connection. Check the callback function.

### **1.1.429. Buffer overflow during protocol decoding, aborting session**

#### **telnet.violation 1**

This message indicates that the Telnet protocol contained a sequence of protocol elements that would have needed an unreasonably large buffer to process. These structures are not commonly found in normal Telnet streams but could be used to mount a denial of service attack on a Telnet client or server by a malicious host.

### **1.1.430. Internal error, not enough literals**

#### **imap.error 1**

This message appears, when Zorp cannot find a literal in message rebuilding.



### **1.1.431. Internal error, too many literals**

#### **imap.error 1**

Placeholder imap.c.2

### **1.1.432. Error while writing policy response**

#### **imap.error 2**

Placeholder imap.c.3

### **1.1.433. Error while writing policy response**

#### **imap.error 3**

Placeholder imap.c.4

### **1.1.434. Failed to parse command**

#### **imap.violation 2**

Placeholder imap.c.5

### **1.1.435. Error writing to client**

#### **imap.error 2**

Placeholder imap.c.6

### **1.1.436. Reconstructed request**

#### **imap.debug 6**

Placeholder imap.c.7

### **1.1.437. Illegal tag**

#### **imap.violation 2**

Placeholder imap.c.8

### **1.1.438. Duplicate tag**

#### **imap.violation 2**

Placeholder imap.c.9



### **1.1.439. Command not allowed in this state**

**imap.policy 2**

Placeholder imap.c.10

### **1.1.440. Error parsing command**

**imap.violation 2**

Placeholder imap.c.10

### **1.1.441. Request denied by policy**

**imap.policy 3**

Placeholder imap.c.11

### **1.1.442. Dropping request**

**imap.policy 3**

Placeholder imap.c.12

### **1.1.443. Policy responds to client**

**imap.policy 4**

Placeholder imap.c.13

### **1.1.444. Aborting connection due to request**

**imap.policy 3**

Placeholder imap.c.14

### **1.1.445. Too many pending requests**

**imap.policy 3**

Placeholder imap.c.15

### **1.1.446. Error writing to server**

**imap.error 2**

Placeholder imap.c.16



### 1.1.447. Stray continuation response

#### imap.error 2

This message appear when reading continuation answer from message but Zorp cannot found a request that need continuation message. this probable means unknown capabilities or synchronization problem between server and proxy.

### 1.1.448. Error writing to server

#### imap.error 1

Placeholder imap.c.18

### 1.1.449. Error writing to server

#### imap.error 1

Placeholder imap.c.19

### 1.1.450. Error writing to server

#### imap.error 1

Placeholder imap.c.20

### 1.1.451. Server returned untagged line during authentication exchange, dropping it

#### imap.violation 5

Placeholder imap.c.21

### 1.1.452. Aborting connection due to a response

#### imap.policy 3

Placeholder imap.c.22

### 1.1.453. Dropping response

#### imap.policy 3

Placeholder imap.c.23

### 1.1.454. Invalid response

#### imap.violation 2

Placeholder imap.c.24





### **1.1.455. Error writing to client**

#### **imap.error 1**

Placeholder imap.c.25

### **1.1.456. Response arrived for unknown request**

#### **imap.violation 5**

Placeholder imap.c.26

### **1.1.457. Aborting connection due to a response**

#### **imap.policy 2**

Placeholder imap.c.27

### **1.1.458. Dropping response**

#### **imap.policy 5**

Placeholder imap.c.28

### **1.1.459. Invalid response**

#### **imap.violation 2**

Placeholder imap.c.28

### **1.1.460. Error writing to client**

#### **imap.error 1**

Placeholder imap.c.30

### **1.1.461. Complete response arrived**

#### **imap.response 6**

Placeholder imap.c.31

### **1.1.462. Reconstructed response**

#### **imap.debug 6**

Placeholder imap.c.33

### **1.1.463. Reading literal**

**imap.debug 6**

Placeholder imap.c.34

### **1.1.464. No free literal space**

**imap.policy 3**

This is appears when literals count reach it's maximum.

### **1.1.465. Illegal character**

**imap.violation 3**

This message appears when not allowed character appear in readed line.

### **1.1.466. Illegal literal length form**

**imap.violation 3**

This message means that literal length contains non number characters

### **1.1.467. Literal length attribute not closed**

**imap.violation 3**

This message appears literal length attribute is not complete. The close parenthesis is not found

### **1.1.468. Stacked proxy rejected contents**

**imap.error 2**

This message indicates that the stacked proxy rejected the content and Zorp rejects the response.

### **1.1.469. Policy type is invalid**

**imap.policy 1**

This message indicates that the policy type is invalid for the given request and Zorp aborts the connection. Check the 'request' attribute.

### **1.1.470. Cannot parse stack policy line**

**imap.policy 1**

This message indicates that the stack policy for the given response is invalid and Zorp stacks nothing. Check the 'response\_stack' attribute. It is likely that the parameter for the POP3\_STK\_MIME or POP3\_STK\_DATA is invalid.

### **1.1.471. Cannot parse stack policy line**

#### **imap.policy 1**

This message indicates that the stack policy for the given response is invalid and Zorp stacks nothing. Check the 'response\_stack' attribute. It is likely that the parameter for the POP3\_STK\_POLICY is invalid.

### **1.1.472. Error in policy call**

#### **imap.policy 1**

This message indicates that the callback for the given request policy is invalid and Zorp stacks nothing. Check the 'request' attribute. It is likely that the parameter for the POP3\_STK\_POLICY is invalid.

### **1.1.473. Cannot parse return code**

#### **imap.policy 1**

This message indicates that the returned value of the callback for the given response policy is invalid and Zorp stacks nothing. Check the callback function.

### **1.1.474. Response count for the request reached limit, drop packets**

#### **ldap.policy 6**

Too many SearchResultEntry message arrived for the corresponding request, and the required action is LDAP\_RSP\_DROP

### **1.1.475. Response count for request reached limit, aborting connection**

#### **ldap.policy 3**

Too many SearchResultEntry message arrived for the corresponding request, and the required action is LDAP\_RSP\_ABORT

### **1.1.476. Invalid value for response\_ouerrun\_action, falling back to LDAP\_RSP\_DROP**

#### **ldap.error 3**

response\_ouerrun\_action may contain other values than LDAP\_RSP\_DROP and LDAP\_RSP\_ABORT, in this case the assumed value is LDAP\_RSP\_DROP

### **1.1.477. Policy nesting do deep**

#### **mime.policy 1**



This message attempt when policy level is too deep. Maybe looping.

#### **1.1.478. Policy type invalid**

**mime.policy 1**

This message attempt when zorp cannot parse the policy line.

#### **1.1.479. Cannot parse policy line**

**mime.policy 1**

This message attempt when zorp cannot parse the policy line

#### **1.1.480. Cannot parse policy line**

**mime.policy 1**

This message attempt when zorp cannot parse the policy line

#### **1.1.481. Cannot parse policy line**

**mime.policy 1**

This message attempt when zorp cannot parse the policy line

#### **1.1.482. Cannot parse policy line**

**mime.policy 1**

This message attempt when zorp cannot parse the policy line

#### **1.1.483. Error in policy calling**

**mime.policy 1**

This message attempt when zorp cannot call the given policy function.

#### **1.1.484. Policy does not contain this header, using hard-coded default**

**mime.policy 5**

This message attempt when zorp cannot found what to do with a command.

#### **1.1.485. Policy does not contain this mime-type, using hard-coded default**

**mime.policy 5**

This message attempt when zorp cannot found what to do with a command answer.

**1.1.486. Answer type invalid****mime.policy 1**

This message attempt when zorp cannot parse the policy line.

**1.1.487. Bad policy line****mime.policy 1**

This message attempt when zorp cannot parse policy line.

**1.1.488. Error in policy calling****mime.policy 1**

This message caused by an exeption in python.

**1.1.489. Return code invalid from policy function****mime.policy 1**

This message attempt when zorp cannot parse policy return value.

**1.1.490. Policy does not contain this error, using hard-coded default****mime.policy 5**

This message attempt when zorp cannot found what to do with a command.

**1.1.491. Cannot parse policy line****mime.policy 1**

This message attempt when zorp cannot parse the policy line

**1.1.492. Error in policy calling****mime.policy 1**

This message attempt when zorp cannot call the given policy function.

**1.1.493. Error parsing return code****mime.policy 1**

This message attempt when called policy function return with wrong value.



### 1.1.494. Stack policy type is invalid

#### **nntp.policy 1**

This message indicates that the stack policy type is invalid for the given response, so nothing will be stacked. Check the 'response\_stack' attribute.

### 1.1.495. Cannot parse stack policy line

#### **nntp.policy 1**

This message indicates that the stack policy for the given response is invalid and Zorp stacks nothing. Check the 'response\_stack' attribute. It is likely that the parameter for the NNTP\_STK\_POLICY is invalid.

### 1.1.496. Error in policy call

#### **nntp.policy 1**

This message indicates that the callback for the given request policy is invalid and Zorp stacks nothing. Check the 'request' attribute. It is likely that the parameter for the NNTP\_STK\_POLICY is invalid.

### 1.1.497. Cannot parse return code

#### **nntp.policy 1**

This message indicates that the returned value of the callback for the given response policy is invalid and Zorp stacks nothing. Check the callback function.

### 1.1.498. Stacked proxy rejected contents

#### **nntp.error 2**

This message indicates that the stacked proxy rejected the content and Zorp rejects the response.

### 1.1.499. Invalid stacking type

#### **ssh.policy 3**

This message indicates that the client\_channel\_stack or the server\_channel\_stack hash contains an invalid stacking type. Check your Zorp configuration.

### 1.1.500. Invalid stacking tuple returned by policy

#### **ssh.policy 3**

This message indicates that the client\_channel\_stack or the server\_channel\_stack hash contains an invalid stacking tuple. It should contain a (stack\_type, proxy\_class) tuple. Check your Zorp configuration.



### 1.1.501. Evaluating BACL rule

**core.auth 5**

This message reports that a REQUIRED type Basic ACL rule is evaluated with the given result.

### 1.1.502. BACL result

**core.auth 4**

This message reports that every Basic ACL rule were evaluated with the given final result.

### 1.1.503. Cache over shift-threshold, shifting

**core.message 3**

This message indicates that the cache size(threshold) is reached, and cache is shifted. @see: Cache.ShiftCache

### 1.1.504. Server connection failure

**core.session 3**

This message indicates that the connection to the server failed.

### 1.1.505. Server connection established

**core.session 3**

This message indicates that the connection to the server succeeded.

### 1.1.506. Server connection failure, no destination

**core.session 3**

This message indicates that the connection to the server can not be established, because no server address is set.

### 1.1.507. All destinations are down, clearing cache and trying again

**core.message 4**

This message reports that the remote end is down and Zorp stores the down state of the remote end, so Zorp wont try to connect to it within the timeout latter.

### 1.1.508. Destination is down, skipping

**core.message 4**



This message reports that the remote end is down, but Zorp does not store the down state of the remote end, so Zorp will try to connect to it next time.

### **1.1.509. Destination is down, keeping state**

#### **core.message 4**

This message reports that the remote end is down and Zorp stores the down state of the remote end, so Zorp wont try to connect to it within the timeout latter.

### **1.1.510. Side stacking failed, socketPair failed**

#### **core.session 3**

This message indicates that side stacking failed, because Zorp was unable to create a socketPair. It is likely that there is now resource available. Try increase fd limits.

### **1.1.511. Side-stacking proxy instance**

#### **core.session 4**

This message indicates that side stacking was successful.

### **1.1.512. Side-stacking failed**

#### **core.error 3**

This message indicates that side stacking failed.

### **1.1.513. Unable to determine service, KZorp service lookup failed**

#### **core.policy 0**

This message indicates that the KZorp result lookup has failed for this session.

### **1.1.514. No applicable service found for this client zone (cached)**

#### **core.policy 2**

This message indicates that no applicable service was found for this client zone in the services cache. It is likely that there is no applicable service configured in this ZoneDispatcher at all. Check your ZoneDispatcher service configuration. @see: Dispatcher.ZoneDispatcher

### **1.1.515. No applicable service found for this client zone**

#### **core.policy 2**

This message indicates that no applicable service was found for this client zone. Check your ZoneDispatcher service configuration. @see: Dispatcher.ZoneDispatcher





### **1.1.516. No applicable service found for this client & server zone (cached)**

**core.policy 2**

This message indicates that no applicable service was found for this client zone in the services cache. It is likely that there is no applicable service configured in this CSZoneDispatcher at all. Check your CSZoneDispatcher service configuration. @see: Dispatcher.CSZoneDispatcher

### **1.1.517. No applicable service found for this client & server zone**

**core.policy 2**

This message indicates that no applicable service was found for this client zone. Check your CSZoneDispatcher service configuration. @see: Dispatcher.CSZoneDispatcher

### **1.1.518. Matching regexp found**

**core.policy 4**

This message reports that a matching regexp pattern was found for the given string.

### **1.1.519. Error opening match file**

**core.policy 3**

This message indicates that the Zorp was unable to open the file containing the match regexps. It is likely that the file does not exist or Zorp is not permitted to read it. @see: Matcher.RegexpFileMatcher

### **1.1.520. Error opening ignore file**

**core.policy 3**

This message indicates that the Zorp was unable to open the file containing the ignore regexps. It is likely that the file does not exist or Zorp is not permitted to read it. @see: Matcher.RegexpFileMatcher

### **1.1.521. Cached recipient match found**

**core.debug 6**

This message reports that the recipient address has been already checked and Zorp uses the cached information.

### **1.1.522. Recipient validity not cached, trying the direct way**

**core.debug 6**

This message reports that the recipient address has not been already checked and Zorp is going to check it now directly.

### **1.1.523. SMTP sender was rejected, unable to verify user existence**

#### **core.error 3**

This message indicates that the sender address was rejected during the recipient address verify check and Zorp rejects the recipient address.

### **1.1.524. Server accepted recipient**

#### **core.info 5**

This message reports that the recipient address verify was successful and Zorp accepts it.

### **1.1.525. Server rejected recipient**

#### **core.info 4**

This message reports that the recipient address verify was unsuccessful and Zorp rejects it.

### **1.1.526. SMTP error during recipient validity checking**

#### **core.error 3**

This message indicates that an SMTP error occurred during the recipient address verify and Zorp rejects it.

### **1.1.527. Before NAT mapping**

#### **core.debug 4**

This message reports that the NAT type and the old address before the NAT mapping occurs.

### **1.1.528. After NAT mapping**

#### **core.debug 4**

This message reports that the NAT type and the new address after the NAT mapping occurred.

### **1.1.529. Proxy ending**

#### **core.session 5**

This message reports that this proxy instance was ended.

### **1.1.530. Proxy destroy**

#### **core.debug 6**

This message reports that this proxy instance was destroyed and freed.

### **1.1.531. Error while stacking child proxy**

**core.error 2**

This message indicates that an error occurred during child proxy stacking. The stacking failed and the subsession is destroyed.

### **1.1.532. Stacking custom child**

**core.debug 7**

This message reports that Zorp is about to stack a new proxy under the current proxy, as a child proxy.

### **1.1.533. User authentication successful**

**core.auth 3**

This message reports that the user authentication was successful.

### **1.1.534. Starting proxy instance**

**core.session 3**

This message reports that a new proxy instance is started.

### **1.1.535. Ending proxy instance**

**core.session 4**

This message reports that a new proxy instance is stopped.

### **1.1.536. Outbound connection not permitted**

**core.policy 1**

This message indicates that a service trying to enter to the given zone was denied by the policy. Check that the destination zone is included in the target zone list of the service.

### **1.1.537. Error resolving hostname**

**core.error 3**

This message indicates that the given hostname could not be resolved. It could happen if the hostname is invalid or nonexistent, or if your resolve setting are not well configured. Check your `"/etc/resolv.conf"`

### **1.1.538. Instance definition not found in policy**

**core.error 0**



This message indicates that the initialization function of the given instance was not found in the policy file.

### **1.1.539. Error downloading KZorp configuration, Python traceback follows**

**core.error 0**

This message indicates that downloading the necessary information to the kernel-level KZorp subsystem has failed.

### **1.1.540. Deinitialization requested for instance**

**core.debug 6**

This message reports that the given instance is stopping.

### **1.1.541. Cleaning up instance**

**core.debug 6**

This message reports that the given instance is freeing its external resources (for example its kernel-level policy objects).

### **1.1.542. Error flushing KZorp configuration**

**core.error 0**

This message indicates that flushing the instance-related information in the kernel-level KZorp subsystem has failed.

### **1.1.543. Client foreign port below 1024**

**ftp.policy 3**

This message indicates that the remote port is below 1024 and due to the violation Zorp is closing connection.

### **1.1.544. Server foreign port below 1024 in passive mode**

**ftp.policy 3**

This message indicates that the remote port is below 1024 and due to the violation Zorp is closing connection.

### **1.1.545. Server foreign port is not good in active mode**

**ftp.policy 3**

This message indicates that the server's remote port is not control\_port-1 or 20 and due to the violation Zorp is closing connection.



### **1.1.546. Unknown side when calling bounceCheck**

**ftp.policy 3**

This message indicates an internal error, please contact the Balasys QA team (devel@balasys.hu).

### **1.1.547. Http accounting**

**http.accounting 4**

This is an accounting message that reports request details.

### **1.1.548. Request administratively prohibited**

**http.request 6**

This message indicates that the request was blocked by the URIFilter.

### **1.1.549. Sender address administratively prohibited**

**smtp.policy 3**

This message indicates that the sender address was administratively prohibited and Zorp rejects the request. Check the 'sender\_matcher' attribute.

### **1.1.550. Sender check successful**

**smtp.debug 6**

This message reports that the sender address check was successful and Zorp accepts the request.

### **1.1.551. Forbidden percent found in address local-part**

**smtp.policy 3**

This message indicates that the email address local-part contains a percent sign and it is not permitted by the policy and Zorp rejects the request. Check the 'permit\_percent\_hack' attribute.

### **1.1.552. Forbidden exclamation mark found in address local-part**

**smtp.policy 3**

This message indicates that the email address local-part contains a exclamation mark and it is not permitted by the policy and Zorp rejects the request. Check the 'permit\_exclamation\_mark' attribute.

### **1.1.553. Relaying denied**

**smtp.policy 3**



This message indicates that relaying the given address is not permitted by the policy and Zorp rejects the request. Check the 'relay\_check' attribute.

### **1.1.554. Relay check successful**

**smtp.debug 6**

This message reports that the relay check was successful and Zorp accepts the request.

### **1.1.555. Recipient address administratively prohibited**

**smtp.policy 3**

This message indicates that the given recipient address is administratively prohibited and Zorp rejects the request. Check the 'recipient\_matcher' attribute.

### **1.1.556. Recipient check successful**

**smtp.debug 6**

This message reports that the recipient check was successful and Zorp accepts the request.

### **1.1.557. Relay check, checking client\_zone**

**smtp.debug 7**

This message reports that Zorp checks the zone of the client.

### **1.1.558. Relay check, checking mail domain**

**smtp.debug 7**

This message reports that Zorp checks the domain name of the email.

### **1.1.559. Signal received, stackdump follows**

**core.error 0**

This message is logged when Zorp caught a fatal signal. Possible reason is bad RAM or other hardware.

### **1.1.560. Invalid license file**

**core.license 0**

This message indicates that Zorp found no or invalid license file. Check your license file at "/etc/zorp/license.txt".

### **1.1.561. Error initializing Python policy engine**

**core.error 0**



This message indicates that Zorp was unable to initialize the Python engine. It is likely that your installation is broken. Check your packages and their version number.



## Chapter 2. Management Server log messages

### 2.1.1. Module successfully loaded

**core.debug 7**

This message serves informational purposes, and indicates that the given module was successfully loaded from the given shared object.

### 2.1.2. Module loading failed

**core.error 1**

This message indicates that loading a module failed.

### 2.1.3. Module loading failed

**core.error 1**

This message indicates that configdb module not loaded.

### 2.1.4. Module loading failed

**core.error 1**

This message indicates that keymanagement module not loaded.

### 2.1.5. Module loading failed

**core.error 1**

This message indicates that transfer module not loaded.

### 2.1.6. Cannot open plugin directory

**core.error 1**

This message indicates that loading a proxy module failed.

### 2.1.7. Module loading failed

**configdb.error 1**

This message indicates that loading a module failed.





### 2.1.8. Cannot open plugin directory

**configdb.error 2**

This message indicates that parsing plugin directory failed.

### 2.1.9. Signal received, stackdump follows

**core.error 0**

This message is logged when ZMS caught a fatal signal. Possible reason is bad RAM or other hardware.

### 2.1.10. SIGHUP received, reloading config

**core.debug 4**

This message is logged when ZMS caught a fatal signal. Possible reason is bad RAM or other hardware.



## Chapter 3. MS Transfer Agent log messages

### 3.1.1. Signal received, stackdump follows

**core.error 0**

This message is logged when agent caught a fatal signal. Possible reason is bad RAM or other hardware.



## Chapter 4. Authentication Server log messages

### 4.1.1. Signal received, stackdump follows

**core.error 0**

This message is logged when ZAS caught a fatal signal. Possible reason is bad RAM or other hardware.

### 4.1.2. Module successfully loaded

**core.debug 6**

This message serves informational purposes, and indicates that the given module was successfully loaded from the given shared object.

### 4.1.3. Module initialization failed

**core.error 3**

This message indicates that loading a module failed.

### 4.1.4. Error dlopening module

**core.info 3**

This message indicates that loading a module failed.

### 4.1.5. Cannot open plugin directory

**core.error 1**

This message indicates that the module directory does not exist.



## Chapter 5. Authentication Agent log messages

### 5.1.1. Signal received, stackdump follows

**core.error 0**

This message is logged when agent caught a fatal signal. Possible reason is bad RAM or other hardware.

### 5.1.2. Signal received, stackdump follows

**core.error 0**

This message is logged when agent caught a fatal signal. Possible reason is bad RAM or other hardware.



## Chapter 6. Content Filtering log messages

### 6.1.1. Module successfully loaded

**core.debug 6**

This message serves informational purposes, and indicates that the given module was successfully loaded from the given shared object.

### 6.1.2. Module initialization failed

**core.error 3**

This message indicates that loading a module failed.

### 6.1.3. Error dlopening module

**core.info 3**

This message indicates that loading a module failed.

### 6.1.4. Cannot open plugin directory

**core.error 1**

This message indicates that the module directory does not exist.

### 6.1.5. Signal received, stackdump follows

**core.error 0**

This message is logged when Zorp caught a fatal signal. Possible reason is bad RAM or other hardware.

# Appendix A. Creative Commons Attribution Non-commercial No Derivatives (by-nc-nd) License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED. BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. TO THE EXTENT THIS LICENSE MAY BE CONSIDERED TO BE A CONTRACT, THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

## 1. Definitions

- a. "Adaptation" means a work based upon the Work, or upon the Work and other pre-existing works, such as a translation, adaptation, derivative work, arrangement of music or other alterations of a literary or artistic work, or phonogram or performance and includes cinematographic adaptations or any other form in which the Work may be recast, transformed, or adapted including in any form recognizably derived from the original, except that a work that constitutes a Collection will not be considered an Adaptation for the purpose of this License. For the avoidance of doubt, where the Work is a musical work, performance or phonogram, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered an Adaptation for the purpose of this License.
- b. "Collection" means a collection of literary or artistic works, such as encyclopedias and anthologies, or performances, phonograms or broadcasts, or other works or subject matter other than works listed in Section 1(f) below, which, by reason of the selection and arrangement of their contents, constitute intellectual creations, in which the Work is included in its entirety in unmodified form along with one or more other contributions, each constituting separate and independent works in themselves, which together are assembled into a collective whole. A work that constitutes a Collection will not be considered an Adaptation (as defined above) for the purposes of this License.
- c. "Distribute" means to make available to the public the original and copies of the Work through sale or other transfer of ownership.
- d. "Licensor" means the individual, individuals, entity or entities that offer(s) the Work under the terms of this License.
- e. "Original Author" means, in the case of a literary or artistic work, the individual, individuals, entity or entities who created the Work or if no individual or entity can be identified, the publisher; and in addition (i) in the case of a performance the actors, singers, musicians, dancers, and other persons who act, sing, deliver, declaim, play in, interpret or otherwise perform literary or artistic works or expressions of folklore; (ii) in the case of a phonogram the producer being the person or legal entity who first fixes the sounds of a performance or other sounds; and, (iii) in the case of broadcasts, the organization that transmits the broadcast.

- f. "Work" means the literary and/or artistic work offered under the terms of this License including without limitation any production in the literary, scientific and artistic domain, whatever may be the mode or form of its expression including digital form, such as a book, pamphlet and other writing; a lecture, address, sermon or other work of the same nature; a dramatic or dramatico-musical work; a choreographic work or entertainment in dumb show; a musical composition with or without words; a cinematographic work to which are assimilated works expressed by a process analogous to cinematography; a work of drawing, painting, architecture, sculpture, engraving or lithography; a photographic work to which are assimilated works expressed by a process analogous to photography; a work of applied art; an illustration, map, plan, sketch or three-dimensional work relative to geography, topography, architecture or science; a performance; a broadcast; a phonogram; a compilation of data to the extent it is protected as a copyrightable work; or a work performed by a variety or circus performer to the extent it is not otherwise considered a literary or artistic work.
  - g. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
  - h. "Publicly Perform" means to perform public recitations of the Work and to communicate to the public those public recitations, by any means or process, including by wire or wireless means or public digital performances; to make available to the public Works in such a way that members of the public may access these Works from a place and at a place individually chosen by them; to perform the Work to the public by any means or process and the communication to the public of the performances of the Work, including by public digital performance; to broadcast and rebroadcast the Work by any means including signs, sounds or images.
  - i. "Reproduce" means to make copies of the Work by any means including without limitation by sound or visual recordings and the right of fixation and reproducing fixations of the Work, including storage of a protected performance or phonogram in digital form or other electronic medium.
2. *Fair Dealing Rights.* Nothing in this License is intended to reduce, limit, or restrict any uses free from copyright or rights arising from limitations or exceptions that are provided for in connection with the copyright protection under copyright law or other applicable laws.
3. *License Grant.* Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:
- a. to Reproduce the Work, to incorporate the Work into one or more Collections, and to Reproduce the Work as incorporated in the Collections; and,
  - b. to Distribute and Publicly Perform the Work including as incorporated in Collections.
- The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats, but otherwise you have no rights to make Adaptations. Subject to 8(f), all rights not expressly granted by Licensor are hereby reserved, including but not limited to the rights set forth in Section 4(d).
4. *Restrictions.* The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

- a. You may Distribute or Publicly Perform the Work only under the terms of this License. You must include a copy of, or the Uniform Resource Identifier (URI) for, this License with every copy of the Work You Distribute or Publicly Perform. You may not offer or impose any terms on the Work that restrict the terms of this License or the ability of the recipient of the Work to exercise the rights granted to that recipient under the terms of the License. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties with every copy of the Work You Distribute or Publicly Perform. When You Distribute or Publicly Perform the Work, You may not impose any effective technological measures on the Work that restrict the ability of a recipient of the Work from You to exercise the rights granted to that recipient under the terms of the License. This Section 4(a) applies to the Work as incorporated in a Collection, but this does not require the Collection apart from the Work itself to be made subject to the terms of this License. If You create a Collection, upon notice from any Licensor You must, to the extent practicable, remove from the Collection any credit as required by Section 4(c), as requested.
- b. You may not exercise any of the rights granted to You in Section 3 above in any manner that is primarily intended for or directed toward commercial advantage or private monetary compensation. The exchange of the Work for other copyrighted works by means of digital file-sharing or otherwise shall not be considered to be intended for or directed toward commercial advantage or private monetary compensation, provided there is no payment of any monetary compensation in connection with the exchange of copyrighted works.
- c. If You Distribute, or Publicly Perform the Work or Collections, You must, unless a request has been made pursuant to Section 4(a), keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or if the Original Author and/or Licensor designate another party or parties (for example a sponsor institute, publishing entity, journal) for attribution ("Attribution Parties") in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; (ii) the title of the Work if supplied; (iii) to the extent reasonably practicable, the URI, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work. The credit required by this Section 4(c) may be implemented in any reasonable manner; provided, however, that in the case of a Collection, at a minimum such credit will appear, if a credit for all contributing authors of Collection appears, then as part of these credits and in a manner at least as prominent as the credits for the other contributing authors. For the avoidance of doubt, You may only use the credit required by this Section for the purpose of attribution in the manner set out above and, by exercising Your rights under this License, You may not implicitly or explicitly assert or imply any connection with, sponsorship or endorsement by the Original Author, Licensor and/or Attribution Parties, as appropriate, of You or Your use of the Work, without the separate, express prior written permission of the Original Author, Licensor and/or Attribution Parties.
- d. For the avoidance of doubt:
  - i. Non-waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme cannot be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights granted under this License;
  - ii. Waivable Compulsory License Schemes. In those jurisdictions in which the right to collect royalties through any statutory or compulsory licensing scheme can be waived, the Licensor reserves the exclusive right to collect such royalties for any exercise by You of the rights



granted under this License if Your exercise of such rights is for a purpose or use which is otherwise than noncommercial as permitted under Section 4(b) and otherwise waives the right to collect royalties through any statutory or compulsory licensing scheme; and,

- iii. Voluntary License Schemes. The Licensor reserves the right to collect royalties, whether individually or, in the event that the Licensor is a member of a collecting society that administers voluntary licensing schemes, via that society, from any exercise by You of the rights granted under this License that is for a purpose or use which is otherwise than noncommercial as permitted under Section 4(b).
  - e. Except as otherwise agreed in writing by the Licensor or as may be otherwise permitted by applicable law, if You Reproduce, Distribute or Publicly Perform the Work either by itself or as part of any Collections, You must not distort, mutilate, modify or take other derogatory action in relation to the Work which would be prejudicial to the Original Author's honor or reputation.
5. *Representations, Warranties and Disclaimer* UNLESS OTHERWISE MUTUALLY AGREED BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.
6. *Limitation on Liability*. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
7. *Termination*
- a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Collections from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
  - b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.
8. *Miscellaneous*
- a. Each time You Distribute or Publicly Perform the Work or a Collection, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
  - b. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further

action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

- c. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
- d. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
- e. The rights granted under, and the subject matter referenced, in this License were drafted utilizing the terminology of the Berne Convention for the Protection of Literary and Artistic Works (as amended on September 28, 1979), the Rome Convention of 1961, the WIPO Copyright Treaty of 1996, the WIPO Performances and Phonograms Treaty of 1996 and the Universal Copyright Convention (as revised on July 24, 1971). These rights and subject matter take effect in the relevant jurisdiction in which the License terms are sought to be enforced according to the corresponding provisions of the implementation of those treaty provisions in the applicable national law. If the standard suite of rights granted under applicable copyright law includes additional rights not granted under this License, such additional rights are deemed to be included in the License; this License is not intended to restrict the license of any rights under applicable law.