

HCP tools コマンドリファレンス

目次

1	コマンドリファレンス	19
1.1	hcpd	19
1.1.1	f, foreground	19
1.1.2	V, version	19
1.1.3	h, help	20
1.1.4	t, config-test	20
1.1.5	q, quit	23
1.1.6	c, config-file	23
1.1.7	l, log-file	24
1.1.8	L, stat-log-file	24
1.1.9	p, pid-file	25
1.1.10	k, license	25
1.1.11	investigation	26
1.2	hcpd_winserv	26
1.2.1	バージョン	27
1.2.2	設定ファイル	27
1.2.3	ログファイル	27
1.2.4	統計ログ	27
1.2.5	ライセンス	27
1.3	hcp	27
1.3.1	p, permission	30
1.3.2	R, recursive	30
1.3.3	Y, anydirs	31
1.3.4	g, regex	31

1.3.5	y, verify	31
1.3.6	z, compress	32
1.3.7	s, copy-linkfile	32
1.3.8	S, follow-linkdir	32
1.3.9	e, no-emptyfile	33
1.3.10	E, no-emptydir	33
1.3.11	d, no-dotfile	33
1.3.12	D, no-dotdir	33
1.3.13	l, copy-hidden.....	34
1.3.14	A, archive-check	34
1.3.15	r, resume	34
1.3.16	N, no-send.....	35
1.3.17	n, no-diskio.....	35
1.3.18	hpf	36
1.3.19	wss.....	36
1.3.20	V, version	37
1.3.21	config-test.....	37
1.3.22	h, help	40
1.3.23	m, copy-mode.....	40
1.3.24	o, overwrite.....	41
1.3.25	a, fail-action.....	41
1.3.26	user.....	42
1.3.27	password.....	42
1.3.28	f, source-file.....	43
1.3.29	—source-file-host.....	44

1.3.30	port.....	44
1.3.31	config-file	45
1.3.32	log-file	45
1.3.33	stat-log-file	46
1.3.34	hcp-out.....	46
1.3.35	multi-run.....	49
1.3.36	overwrite-als-temp-file	50
1.3.37	preserve-win-read-only	51
1.3.38	udp.....	51
1.3.39	ws.....	52
1.3.40	wss-no-check-certificate.....	53
1.3.41	ws-proxy	53
1.3.42	hfp-cong.....	53
1.3.43	hfp-mss.....	54
1.3.44	hfp-sndbuf.....	54
1.3.45	hfp-rcvbuf	55
1.3.46	reading-dev	55
1.3.47	reading-dev-via-mmap	55
1.3.48	transfer-expire.....	55
1.3.49	no-out-of-order-header-messaging.....	56
1.3.50	mcd.....	56
1.3.51	auto-resume.....	56
1.4	hsync (実験的)	57
1.4.1	v, verbose	61
1.4.2	q, quiet.....	61

1.4.3	c, checksum	62
1.4.4	a, archive	62
1.4.5	r, recursive.....	62
1.4.6	b, backup	62
1.4.7	backup-dir	63
1.4.8	suffix.....	63
1.4.9	u, update.....	63
1.4.10	d, dirs.....	64
1.4.11	l, links	64
1.4.12	L, copy-links.....	65
1.4.13	copy-unsafe-links	65
1.4.14	safe-links	65
1.4.15	k, copy-dirlinks	66
1.4.16	p, perms.....	66
1.4.17	E, executability.....	66
1.4.18	chmod.....	67
1.4.19	o, owner.....	67
1.4.20	g, group	68
1.4.21	t, times.....	68
1.4.22	O, omit-dir-times.....	68
1.4.23	J, omit-link-times	69
1.4.24	dry-run.....	69
1.4.25	existing.....	69
1.4.26	ignore-existing.....	70
1.4.27	del	70

1.4.28	delete	70
1.4.29	delete-during.....	71
1.4.30	force	71
1.4.31	max-delete.....	71
1.4.32	max-size.....	72
1.4.33	min-size	72
1.4.34	m, prune-empty-dirs	73
1.4.35	numeric-ids	73
1.4.36	usermap.....	73
1.4.37	groupmap.....	74
1.4.38	chown	74
1.4.39	l, ignore-times	75
1.4.40	size-only	75
1.4.41	z, compress.....	76
1.4.42	compress-level.....	76
1.4.43	f, filter	76
1.4.44	exclude	77
1.4.45	exclude-from	78
1.4.46	include.....	78
1.4.47	include-from	79
1.4.48	log-file	79
1.4.49	bwlimit.....	80
1.4.50	stop-after	80
1.4.51	-V, version.....	81
1.4.52	config-test.....	81

1.4.53	h, help	85
1.4.54	hcp-log-file	85
1.4.55	hcp-stat-log-file	85
1.5	hrm.....	85
1.5.1	f, force.....	87
1.5.2	R, recursive.....	87
1.5.3	d, dir	87
1.5.4	i	88
1.5.5	l	88
1.5.6	V, version.....	88
1.5.7	config-test	89
1.5.8	h, help	91
1.5.9	host.....	91
1.5.10	port.....	92
1.5.11	hcp-out.....	92
1.6	hcp-ls.....	93
1.6.1	q, query-cmdname.....	94
1.6.2	o, cmd-options.....	94
1.6.3	V, version.....	95
1.6.4	config-test	95
1.6.5	h, help	97
1.6.6	host.....	97
1.6.7	port.....	98
1.6.8	hcp-out.....	98
1.7	hmkdir	99

1.7.1	p, parents.....	100
1.7.2	V, version.....	100
1.7.3	config-test.....	101
1.7.4	h, help.....	103
1.7.5	host.....	103
1.7.6	port.....	104
1.7.7	hcp-out.....	104
1.8	hpwd.....	105
1.8.1	L, logical.....	106
1.8.2	P, physical.....	106
1.8.3	V, version.....	106
1.8.4	config-test.....	107
1.8.5	h, help.....	109
1.9	hmv.....	109
1.9.1	f, force.....	111
1.9.2	i, interactive.....	111
1.9.3	N, no-overwrite.....	111
1.9.4	V, version.....	111
1.9.5	config-test.....	112
1.9.6	h, help.....	114
1.9.7	host.....	114
1.9.8	port.....	115
1.9.9	hcp-out.....	115
1.10	hln.....	116
1.10.1	f, force.....	117

1.10.2	i, interactive	118
1.10.3	N, no-dereference	118
1.10.4	s, symbolic	118
1.10.5	L, logical	119
1.10.6	P, physical	119
1.10.7	V, version	119
1.10.8	config-test	120
1.10.9	h, help	122
1.10.10	host	122
1.10.11	port	122
1.10.12	hcp-out	123
1.11	hchmod	123
1.11.1	R, recursive	125
1.11.2	V, version	125
1.11.3	config-test	125
1.11.4	h, help	128
1.11.5	host	128
1.11.6	port	128
1.11.7	hcp-out	129
1.12	hchown	129
1.12.1	d, no-dereference	131
1.12.2	R, recursive	131
1.12.3	s, follow-cmd-link-dir	131
1.12.4	S, follow-all	132
1.12.5	D, no-follow	132

1.12.6	V, version	132
1.12.7	config-test.....	133
1.12.8	h, help	135
1.12.9	host.....	135
1.12.10	port.....	135
1.12.11	hcp-out.....	136
2	設定リファレンス	136
2.1	hcpd.conf	136
2.1.1	TCPListenAddress	140
2.1.2	TCPServiceSocketSendBuffer.....	142
2.1.3	HPFPListenAddress	142
2.1.4	UDPListenAddress.....	144
2.1.5	UDPServiceExtensionBufferSize	146
2.1.6	WSSListenAddress	146
2.1.7	WSSOptions.....	149
2.1.8	WSSCipherSuites.....	149
2.1.9	WSSCipherList.....	150
2.1.10	WSListenAddress.....	151
2.1.11	ListenServiceBonding	152
2.1.12	UseServerCertificateSecurity	153
2.1.13	RequireServerCertificateSecurity	153
2.1.14	ServerKeyFile	153
2.1.15	ServerCertificateFile.....	154
2.1.16	ServerCertificateChainFile	155
2.1.17	LocalPasswordAuthentication.....	156

2.1.18	PAMAuthentication.....	156
2.1.19	PubkeyAuthentication.....	157
2.1.20	WinLogonUserAuthentication.....	157
2.1.21	PerformSystemAuthenticationRegardlessUsers.....	157
2.1.22	UserDirectoryFallbackAvailable.....	158
2.1.23	RejectOnUserHomeDirectoryNotFound.....	158
2.1.24	UsePrivilegeSeparation.....	159
2.1.25	PrivilegeSeparationMinimumUID.....	159
2.1.26	PrivilegeSeparationMinimumGID.....	160
2.1.27	PrivilegeSeparationUser.....	160
2.1.28	PrivilegeSeparationUmask.....	161
2.1.29	PrivilegeSeparationUmaskAnonymous.....	162
2.1.30	ApplyUserPermission.....	162
2.1.31	AllowUsers.....	163
2.1.32	AllowGroups.....	164
2.1.33	DenyUsers.....	164
2.1.34	DenyGroups.....	165
2.1.35	AuthorizedKeySearchDir.....	166
2.1.36	AuthorizedKeyFile.....	167
2.1.37	CACertificateFile.....	167
2.1.38	CACertificatePath（予約）.....	167
2.1.39	CARevocationFile.....	168
2.1.40	CARevocationPath（予約）.....	168
2.1.41	OCSPRevocationEnabled.....	169
2.1.42	LocalUserFile.....	169

2.1.43	LocalPasswordFile	170
2.1.44	AcceptableCryptMethod	171
2.1.45	AcceptableDigestMethod.....	171
2.1.46	RequireDataIntegrityChecking.....	172
2.1.47	TransportCharEncoding	173
2.1.48	HostEncoding.....	173
2.1.49	HeaderCompress（予約）	174
2.1.50	ContentCompress（予約）	174
2.1.51	MaxConcurrentThread	174
2.1.52	MaxTotalConnection	174
2.1.53	MaxTcpConnection	174
2.1.54	MaxUdpConnection.....	175
2.1.55	MaxWsConnection.....	175
2.1.56	MaxConnectionPerUser.....	176
2.1.57	MaxConnectionPerSec	176
2.1.58	MaxReceiveFileSize.....	176
2.1.59	MaxSendFileSize	177
2.1.60	MaxRequestFileEntryAtOnce.....	177
2.1.61	MaxTotalBufferSize.....	177
2.1.62	MaxBufferSizePerConnection	178
2.1.63	MaxTotalReceiveRate.....	178
2.1.64	MaxTotalSendRate	179
2.1.65	MaxReceiveRatePerConnection	179
2.1.66	MaxSendRatePerConnection.....	179
2.1.67	InitHeaderBlockSize	180

2.1.68	InitContentBlockSize	180
2.1.69	MaxHeaderBlockSize	181
2.1.70	MaxContentBlockSize	181
2.1.71	FileLock	182
2.1.72	FileLockTrials	182
2.1.73	FileLockTrialInterval	183
2.1.74	AtomicLikeSaving	183
2.1.75	AtomicLikeSavingThreshold	184
2.1.76	AtomicLikeSavingRejectOverwriteRequest	185
2.1.77	TransportTimeout	185
2.1.78	IdleTimeout	185
2.1.79	DocPoint	186
2.1.80	DocPath	186
2.1.81	PermitAccessRead	187
2.1.82	PermitAccessWrite	187
2.1.83	PermitAccessOverwrite	188
2.1.84	PermitAccessDelete	188
2.1.85	PermitAccessRandomRead (予約)	189
2.1.86	PermitAccessRandomWrite (予約)	189
2.1.87	AccessList	189
2.1.88	Allow	190
2.1.89	Deny	191
2.1.90	SyslogOption	191
2.1.91	SyslogFacility	192
2.1.92	SystemLog	192

2.1.93	SystemLogLevel.....	195
2.1.94	ApplicationStatLog.....	196
2.1.95	TransportStatLog.....	197
2.1.96	SystemStatLog.....	198
2.1.97	FileOperationLog.....	200
2.1.98	CallbackScript	204
2.1.99	MemoryTransferConcurrency	208
2.1.100	MaxReadRatePerConnection.....	209
2.1.101	MaxWriteRatePerConnection	209
2.1.102	EnsureDestinationInFileTransfer	210
2.1.103	StatLogPerUserInPrivilegeSeparation	210
2.1.104	ApplicationStatLogSecurityEx.....	210
2.2	クライアントコマンド共通設定項目	211
2.2.1	RequireServerCertificateSecurity.....	213
2.2.2	RejectFallbackServerCertificateSecurity	214
2.2.3	IgnoreCertificateCNInvalid	214
2.2.4	IgnoreCertificateDateInvalid.....	214
2.2.5	IgnoreUnknownCA.....	215
2.2.6	IgnoreRevocation.....	215
2.2.7	WSSIgnoreCertificateCNInvalid	216
2.2.8	WSSIgnoreCertificateDateInvalid.....	216
2.2.9	WSSIgnoreUnknownCA.....	216
2.2.10	WSSIgnoreRevocation (予約)	217
2.2.11	WSSOptions.....	217
2.2.12	WSSCipherSuites.....	217

2.2.13	WSSCiperList	218
2.2.14	CACertificateFile	219
2.2.15	CACertificatePath (予約)	219
2.2.16	CARevocationFile.....	220
2.2.17	CARevocationPath (予約)	220
2.2.18	WSSCACertificateFile	221
2.2.19	OCSPRevocationEnabled	221
2.2.20	StrictHostKeyChecking.....	221
2.2.21	LocalPasswordAuthentication.....	222
2.2.22	PAMAuthentication.....	222
2.2.23	PubkeyAuthentication	223
2.2.24	WinLogonUserAuthentication.....	223
2.2.25	PrivateKeySearchDir	223
2.2.26	PrivateKeyFile.....	224
2.2.27	AcceptableCryptMethod	225
2.2.28	AcceptableDigestMethod.....	226
2.2.29	DisableDataIntegrityChecking.....	226
2.2.30	AcceptDataIntegrityCheckingOnRejection	227
2.2.31	TransportCharEncoding	227
2.2.32	HostEncoding.....	228
2.2.33	CompressLevel	228
2.2.34	HeaderCompress.....	229
2.2.35	ContentCompress	229
2.2.36	MaxConcurrentThread	230
2.2.37	MaxReceiveFileSize.....	230

2.2.38	MaxSendFileSize	230
2.2.39	MaxRequestFileEntryAtOnce	231
2.2.40	MaxBufferSize	231
2.2.41	MaxReceiveRate	231
2.2.42	MaxSendRate	232
2.2.43	InitHeaderBlockSize	232
2.2.44	InitContentBlockSize	233
2.2.45	MaxHeaderBlockSize	233
2.2.46	MaxContentBlockSize	233
2.2.47	FileLock	234
2.2.48	FileLockTrials	234
2.2.49	FileLockTrialInterval	234
2.2.50	AtomicLikeSaving	234
2.2.51	AtomicLikeSavingThreshold	234
2.2.52	TransportTimeout	234
2.2.53	ApplicationLog	234
2.2.54	ApplicationLogLevel	236
2.2.55	ApplicationStatLog	236
2.2.56	TransportStatLog	237
2.2.57	UDPTransportExtensionBufferSize	238
2.2.58	TCPTransportSocketSendBuffer	239
2.2.59	MemoryTransferConcurrency	239
2.2.60	MaxReadRate	240
2.2.61	MaxWriteRate	241
2.2.62	PubkeyAuthenticationPrior	241

2.2.63	ApplicationStatLogSecurityEx	242
2.3	hcp.conf	242
2.3.1	Include	243
2.3.2	UseProperCopyAndSync	243
2.3.3	AutoResumeTrials	245
2.3.4	AutoResumeTrialInterval	245
2.4	hrm.conf	245
2.4.1	Include	245
2.5	hcp-ls.conf	246
2.5.1	Include	246
2.6	hmkdir.conf	246
2.6.1	Include	246
2.7	hpwd.conf	246
2.7.1	Include	246
2.8	hmv.conf	246
2.8.1	Include	246
2.9	hln.conf	246
2.9.1	Include	246
2.10	users	247
2.11	passwd	247
3	統計リファレンス	248
3.1	アプリケーション統計	248
3.2	トランスポート統計	253
3.2.1	TCP 統計	253
3.2.2	HpFP 統計	255

3.2.3	WS/WSS 統計.....	257
3.3	システム統計.....	259
4	エラーコードリファレンス	260
4.1	コマンド終了ステータス.....	260
4.2	設定ファイルエラーコード.....	261
4.3	ファイル処理理由コード.....	263
4.4	サービスシグナルハンドリング	266
4.5	アプリケーションシグナルハンドリング	267
5	コマンド実行モード	268
5.1	単一起動モード.....	268
5.2	多重起動モード.....	270
6	HCP tools 制約事項一覧.....	272
7	表記法について	272
8	改訂履歴	273

1 コマンドリファレンス

1.1 hcpd

Usage: hcpd [OPTION]...
or : hcpd -q [OPTION]...

hcpd デーモンは、リモート（クライアント）からの hcp（ファイル転送）コマンドや hrm（ファイル削除）コマンドの指示を受け付け、その処理を実行するためにサーバ側で待機するソフトウェアです。Linux 版で提供されます。

短縮名	オプション名	概略
f	foreground	フォアグラウンド実行
V	version	バージョンを表示
h	help	ヘルプを表示
t	config-test	入力パラメータ及び設定ファイルのロード結果を表示する
q	quit	停止シグナルを送信する
c	config-file	設定ファイル
l	log-file	ログファイル
L	stat-log-file	統計ログファイル
p	pid-file	PID ファイル
k	license	ライセンスキーファイル
	investigation	調査モード

1.1.1 f, foreground

```
=====
対応 OS : Linux.x86
書式 : -f | --foreground
=====
```

hcpd デーモンを、フォアグラウンドモードで起動します。

```
--
例 :
[root@localhost ~]# hcpd -f ...
--
```

1.1.2 V, version

```
=====
対応 OS : Linux.x86
```

書式 : -V | --version

=====

hcpd デーモンのバージョンを表示します。

--

例 :

```
[root@localhost ~]# hcpd -V
hcp server (hcpd) 1.3.0_42 / Linux (HpFP2 2.0.0.91_16)
```

--

1.1.3 h, help

=====

対応 OS : Linux.x86

書式 : -h | --help

=====

hcpd デーモンのヘルプを表示します。

--

例 :

```
[root@localhost ~]# hcpd -h
```

--

1.1.4 t, config-test

=====

対応 OS : Linux.x86

書式 : -t | --config-test

=====

hcpd デーモンの入力パラメータ及び設定情報を出力します。

--

例 :

```
[root@localhost ~]# hcpd -t
```

...

Number of Logical Processors : 2

Number of Physical Processors : 2

Command parameters

foreground : disable

version : disable

help : disable

quit : disable

pid-file : disable [/var/run/hcpd.pid] default

license : disable [/etc/hcp/license.key] default

HCP tools コマンドリファレンス

```
config-test : enable
config-file : disable [/etc/hcp/hcpd.conf] default
log-file    : disable [/var/log/hcpd.log] default
stat-log-file : disable [/var/tmp/.hcp.statistics] default
```

Configuration parameters

```
WinLogonUserAuthentication      : no
PubkeyAuthentication           : yes
PAMAuthentication              : yes
LocalPasswordAuthentication    : no
PerformSystemAuthenticationRegardlessUsers : no
UserDirectoryFallbackAvailable : no
RejectOnUserHomeDirectoryNotFound : no
UsePrivilegeSeparation         : yes
PrivilegeSeparationMinimumUID  : 0
PrivilegeSeparationMinimumGID  : 0
PrivilegeSeparationUser        : nobody
PrivilegeSeparationUmask       : 0022 -
PrivilegeSeparationUmaskAnonymous : 0002 -
ApplyUserPermission            : yes
AllowUsers                     : -
AllowGroups                    : -
DenyUsers                     : -
DenyGroups                    : -
UseServerCertificateSecurity   : yes
RequireServerCertificateSecurity : yes
HeaderCompress                 : yes
ContentCompress                : yes
OCSPRevocationEnabled          : yes
AuthorizedKeySearchDir         : - [/etc/hcp/authkeys]
AuthorizedKeyFile               : - [~/.hcp/authorized_keys]
LocalUserFile                  : - [/etc/hcp/users usage=overwrite]
LocalPasswordFile              : - [/etc/hcp/passwd]
ServerKeyFile                  : - [/etc/hcp/key/server.key]
ServerCertificateFile          : - [/etc/hcp/cert/server.crt]
ServerCertificateChainFile     : - [/etc/hcp/cert/chain.crt]
CACertificateFile              : - [/etc/hcp/cacert.pem]
CACertificatePath              : - [/etc/ssl]
CARevocationFile               : - [/etc/hcp/crl.pem]
CARevocationPath               : - [/etc/ssl]
ProtocolVersion                : 2
MaxConcurrentThread            : 0
MaxTotalConnection             : 150
MaxTcpConnection               : 50
MaxUdpConnection               : 50
MaxWsConnection                : 50
MaxConnectionPerUser           : 50
```

HCP tools コマンドリファレンス

```
MaxConnectionPerSec           : 50
MaxRequestFileEntryAtOnce     : 50
MaxReceiveFileSize            : unlimited
MaxSendFileSize               : unlimited
MaxTotalBufferSize           : 4294967296
MaxBufferSizePerConnection    : 104857600
MaxTotalReceiveRate           : 10000000000
MaxTotalSendRate              : 10000000000
MaxReceiveRatePerConnection   : 10000000000
MaxSendRatePerConnection      : 10000000000
InitHeaderBlockSize           : 51200
InitContentBlockSize          : 1048576
MaxHeaderBlockSize            : 51200
MaxContentBlockSize           : 1048576
TransportTimeout              : 180
IdleTimeout                   : 600
FileLock                      : no
FileLockTrials                : 0
FileLockTrialInterval         : 3
AtomicLikeSaving              : no .tmp NONE [threshold=0, reject_ow_
req=yes]
TCPListenAddress              : 0.0.0.0:874[TCP tcp1, mcd=3]
HPFPListenAddress             : 0.0.0.0:65520[HpFP udp1, sndbuf=10485
7600, rcvbuf=209715200, mss=-1, mcd=3]
ListenServiceBonding          : -
UDPServiceExtensionBufferSize : 2147483648
TCPServiceSocketSendBuffer    : 0
AcceptableCryptMethod         : AES256/CTR/VMAC AES256/CBC AES128/CBC
[Intel:AES-NI=yes]
AcceptableDigestMethod        : MM128 SHA256 SHA160
RequireDataIntegrityChecking  : yes
TransportCharEncoding         : UTF8
DocPoint                      : /home
  DocPath = /home
  PermitAccessRead = yes
  PermitAccessWrite = yes
  PermitAccessOverwrite = yes
  PermitAccessDelete = yes
  PermitAccessRandomRead = no
  PermitAccessRandomWrite = no
DocPointEnd
HostEncoding                   : UTF8
SyslogOption                   : LOG_CONS | LOG_PID
SyslogFacility                 : LOG_DAEMON
SystemLog                      : INFO yes Rotation[size=no:0:0, patter
n=no:] -[/var/log/hcpd.log]
ApplicationStatLog             : yes Rotation[size=no:0:0, pattern=n
```

```

o:]
  TransportStatLog           : no Rotation[size=no:0:0, pattern=no:]
  SystemStatLog             : yes Rotation[size=no:0:0, pattern=n
o:]
  FileOperationLog          : no -[/var/log/hcpd.file.operation.lo
g] Rotation[size=no:0:0, pattern=no:]
  CallbackScript            : no Script[~/hcp/callback.sh, data=~/
hcp/callback]
  EnsureDestinationInFileTransfer : yes
  StatLogPerUserInPrivilegeSeparation : no
  ApplicationStatLogSecurityEx   : yes
  MemoryTransferConcurrency     : 1 (wait_type cond, exp_nsec 1)
  MaxReadRatePerConnection      : unlimited
  MaxWriteRatePerConnection     : unlimited
--

```

1.1.5 q, quit

```

=====
対応 OS : Linux.x86
書式 : -q | --quit
=====

```

hcpd デーモンを停止するシグナルを送信します。

pid-file オプションで指定されたファイルもしくは既定のファイルからプロセス ID を取得し、このプロセス ID を指定してシグナルを送信します。

```

--
例 :
[root@localhost ~]# hcpd -q ...
--

```

1.1.6 c, config-file

```

=====
対応 OS : Linux.x86
書式 : -c <config-file-path> | --config-file=<config-file-path>
-----

```

```

config-file-path
既定値 : なし
値の範囲 : ファイルシステムのパス文字列
=====

```

hcpd デーモンが使用する設定ファイルのパスを指定します。

```

--
例 :

```

```
[root@localhost ~]# hcpd -c /etc/hcp/hcpd2.conf ...  
--
```

このオプションが指定されなかった場合は、次のファイルから設定を読み込みます。

/etc/hcp/hcpd.conf

1.1.7 -l, log-file

=====
対応 OS : Linux.x86

書式 : -l <log-file-path> | --log-file=<log-file-path>

log-file-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列
=====

hcpd デーモンのログを指定したパスのファイルに出力します。

--

例 :

```
[root@localhost ~]# hcpd -l /var/log/hcpd.log ...  
--
```

このオプションが指定されなかった場合は、標準出力にログを出力します。

1.1.8 -L, stat-log-file

=====
対応 OS : Linux.x86

書式 : -L <log-file-path> | --stat-log-file=<log-file-path>

log-file-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列
=====

hcpd デーモンが出力する統計ログの基準パスを指定します。

--

例 :

```
[root@localhost ~]# hcpd -L /var/tmp/.hcp.statistics2 ...  
--
```

指定されたパスに接尾辞を付加して各統計ログが出力されます。

<指定されたパス>.system (システム統計)

<指定されたパス>.application (アプリケーション統計)

<指定されたパス>.transport.tcp.service_<サービス番号>.<サービスポート番号>.thread_<スレッド番号> (TCP トランスポート統計)
<指定されたパス>.transport.hpfp.service_<サービス番号>.<サービスポート番号>.thread_<スレッド番号> (HpFP トランスポート統計)
<指定されたパス>.transport.ws.service_<サービス番号>.<サービスポート番号>.thread_<スレッド番号> (WS/WSS トランスポート統計)

非特権ユーザで後述の StatLogPerUserInPrivilegeSeparation が有効(yes)の場合は、次の様なパスに記録されます。

<指定されたパス>.application.<UID>_<GID> (Linux)
<指定されたパス>.application.<Username> (Windows)

指定しない場合は、次のパスに出力します。

/var/tmp/.hcp.statistics

1.1.9 p, pid-file

=====
対応 OS : Linux.x86

書式 : -p <pid-file-path> | --pid-file=<pid-file-path>

pid-file-path

既定値 : /var/run/hcpd.pid

値の範囲 : ファイルシステムのパス文字列
=====

hcpd デーモンプロセスのプロセス ID を出力するファイルパスを指定します。

--

例 :

[root@localhost ~]# hcpd -p /var/run/hcpd2.pid ...

--

複数のデーモンプロセスを起動する場合もしくは既に他のデーモンプロセスが起動している場合は、同じファイルパスを使用しないようご注意ください。

1.1.10 k, license

=====
対応 OS : Linux.x86

書式 : -k <license-key-path> | --license=<license-key-path>

license-key-path

既定値 :

/etc/hcp/license.key

値の範囲 : ファイルシステムのパス文字列

ライセンスキーのパスを指定します。

--

例 :

```
[root@localhost ~]# hcpd -k /etc/hcp/license2.key ...
```

--

指定しない場合は、次のパスからライセンスキーを読み込みます。

`/etc/hcp/license.key`

ライセンスキーが読み込めない場合は、トライアルライセンスが次の要件で適用されます。

合計スループット制限 1Gbps

スループット制限 1Gbps

TCP 最大接続数 3

HpFP 最大接続数 3

WS/WSS 最大接続数 3

最大多重接続数 1

1.1.11 investigation

対応 OS : Linux.x86

書式 : `--investigation`

hcpd デーモンを調査モードで起動します。期待しない動作をする場合などに、調査のために詳細なログを出力します。多量のログが出力されるため、長時間の使用もしくは性能が必要とされるケースでは使用しません。また、取得したログはユーザが可読な内容ではないため、本製品提供元などへ調査目的等のため送付してください。

1.2 hcpd_winserv

hcpd_winserv は、リモート（クライアント）からの hcp（ファイル転送）コマンドや hrm（ファイル削除）コマンドの指示を受け付け、その処理を実行するためにサーバ側で待機するソフトウェアです。Windows 版でサービスプログラムとして提供されます。また、本プログラムはコマンドプロンプトで実行する機能は提供しません。

1.2.1 バージョン

hcpd_winserv のバージョンは、後述のログファイルにサービス起動時に記録されます。

--

例 :

```
hcp server (hcpd) 1.3.0_42 / Windows (HpFP2 2.0.0.91_16)
```

--

1.2.2 設定ファイル

hcpd_winserv は、設定を次のファイルから読み込みます。

```
C:/ProgramData/Clealink/HCP Tools/hcpd.conf
```

1.2.3 ログファイル

アプリケーションのログは次のファイルに書き込まれます。

```
C:/ProgramData/Clealink/HCP Tools/hcpd.log
```

1.2.4 統計ログ

Windows 版では、統計ログは次のパスに出力されます。追加される接尾辞については、前述の Linux 版と同じ文字列が使用されます。

```
C:/ProgramData/Clealink/HCP Tools/Temp/_hcp.statistics
```

1.2.5 ライセンス

次のパスからライセンスキーを読み込みます。

```
C:/ProgramData/Clealink/HCP Tools/license.key
```

ライセンスキーが読み込めない場合は、Linux 版と同じトライアルライセンスが適用されます。

1.3 hcp

```
Local  : hcp [OPTION]... SOURCE [SOURCE]... DEST
        or  : hcp [OPTION]... -f SOURCE_LIST_FILE DEST
```

Remote :

```
Push  : hcp [OPTION]... SOURCE [SOURCE]... [USER@]HOST[:PORT]:DEST
Pull  : hcp [OPTION]... [USER@]HOST[:PORT]:SOURCE [:SOURCE]... DEST
```

Remote using source file list :

```
Push  : hcp [OPTION]... -f SOURCE_LIST_FILE [USER@]HOST[:PORT]:DEST
```

HCP tools コマンドリファレンス

Pull : hcp [OPTION]... -f SOURCE_LIST_FILE DEST

Pull : hcp [OPTION]... --host=HOST [--port=PORT] [--user=USER] -f SOURCE_LIST_FILE DEST

hcp コマンドは、リモート（サーバ、hcpd デーモン）との間でファイル転送を行うコマンドです。ローカルのファイルコピー機能もサポートします。

短縮名	オプション名	概略
p	permission	転送元のパーミッションを保持する
R	recursive	再帰的にコピーする
Y	anydirs	ディレクトリはすべて探索する（ワイルドカード、正規表現の適用から外す）
g	regex	ファイル名の指定に正規表現を使用する（ワイルドカードを使用しない）
y	verify	データをダイジェストにより検証する
z	compress	圧縮する
s	copy-linkfile	シンボリックリンクを解決(Dereference)してファイルをコピーする
S	follow-linkdir	シンボリックリンクのディレクトリを解決してファイルをコピーする
e	no-emptyfile	空のファイルをコピーしない
E	no-emptydir	空のディレクトリをコピーしない
d	no-dotfile	ドットで始まるファイル名のファイルをコピーしない
D	no-dotdir	ドットで始まるディレクトリ名のディレクトリをコピーしない
I	copy-hidden	隠し属性ファイルをコピーする
A	archive-check	アーカイブ属性が設定されているファイルをコピーする
r	resume	再開処理を行う（前回の実行記録が必要）
N	no-send	トランスポート上で送受信はせずディスク I/O 性能を計測する
n	no-diskio	ディスク I/O 操作を行わず通信プロトコルのネットワーク I/O 性能を計測する

	hfpfp	HpFP プロトコルを使用する
	wss	WebSocket プロトコル(SSL/TLS)を使用する
V	version	バージョンを表示する
	config-test	入力パラメータ及び設定ファイルのロード結果を表示する
h	help	コマンドラインヘルプを表示する
m	copy-mode	コピーモード
o	overwrite	上書きモード
a	fail-action	処理エラー時動作指定
	user	ユーザ名
	password	パスワード
f	source-file	ソースファイルリスト
	source-file-host	リモートホスト指定（ソースファイル使用時）
	port	リモートポート指定（ソースファイル使用時）
	config-file	設定ファイル
	log-file	ログファイル
	stat-log-file	統計ログファイル
	hcp-out	実行記録を出力するファイルを指定する
	investigation	調査モード
	multi-run	多重起動モード
	overwrite-als-temp-file	アトミック（不可分）保存 一時ファイル上書き要求
	preserve-win-read-only	Windows 読み込み属性保持
	udp	UDP (HpFP)プロトコルを使用する（ポート分離型）
	ws	WebSocket プロトコル（平文通信）を使用する
	wss-no-check-certificate	WebSocket SSL/TLS サーバ証明書検証無効化
	ws-proxy	WebSocket プロキシサーバ指定
	hfpfp-cong	HpFP プロトコルの輻輳制御モードを指定
	hfpfp-mss	HpFP プロトコルの MSS 指定
	hfpfp-sndbuf	HpFP プロトコルの送信バッファサイズ指定

	hfp-rcvbuf	HpFP プロトコルの送信バッファサイズ指定
	reading-dev	スペシャルデバイス逐次読み取り
	reading-dev-via-mmap	スペシャルデバイス Mapped I/O 読み取り
	transfer-expire	転送期限指定
	no-out-of-order-header-messaging	制御メッセージアウトオブオーダー抑制モード
	mcd	多重接続数指定
	auto-resume	自動再開

次のオプションは hcpd コマンドの説明をご参照ください。

--investigation

1.3.1 p, permission

=====
対応 OS : Linux / Windows

書式 : -p | --permission
=====

転送元のファイルパーミッション (UID/GID, 実行権フラグなど) やファイル属性 (更新日時など) を転送先で保持します。アクセス権などの理由で転送先でパーミッションやファイル属性を更新できない場合は、ファイル書込みを行った際に OS により適用された属性などがそのまま保持されます。

--

例 :

```
[user@localhost ~]$ hcp -p ...
```

--

1.3.2 R, recursive

=====
対応 OS : Linux / Windows

書式 : -R | --recursive
=====

転送元のディレクトリを再帰的に探索してファイル転送を行います。

--

例 :

```
[user@localhost ~]$ hcp -R ...
```

--

1.3.3 Y, anydirs

```
=====
対応 OS : Linux / Windows
書式 : -Y | --anydirs
=====
```

regex オプションやワイルドカードなどによるパターンマッチングが指定された際に、ディレクトリ探索でこのパターンマッチングを無視して探索を行います。

```
--
例 :
[user@localhost ~]$ hcp -Y ...
--
```

1.3.4 g, regex

```
=====
対応 OS : Linux / Windows
書式 : -g | --regex
=====
```

転送元のパスを正規表現として解釈します。このオプションを使用しない場合は、転送元のパスはワイルドカード(*)を含むパターン文字列表現として解釈します。

正規表現（もしくはワイルドカードを含むパターン文字列）による評価は、指定した転送元のパスがディレクトリ及びファイルとしても存在しない場合に行われます。この評価は、ファイル探索時のファイル名及びディレクトリ名に対して適用されます。パターンにマッチしなかったファイルやディレクトリはスキップされます。パターン文字列は、最後のパス区切り文字 (/ など) 以降の文字列から抽出されます。

```
--
例 :
[user@localhost ~]$ hcp -g ...
--
```

1.3.5 y, verify

```
=====
対応 OS : Linux / Windows
書式 : -y | --verify
=====
```

転送するデータブロック及びファイルに対してメッセージダイジェストの検査を行います。

--

例 :

```
[user@localhost ~]$ hcp -y ...
```

--

1.3.6 z, compress

=====

対応 OS : Linux / Windows

書式 : -z | --compress

=====

転送するデータブロックを圧縮します。

--

例 :

```
[user@localhost ~]$ hcp -z ...
```

--

1.3.7 s, copy-linkfile

=====

対応 OS : Linux / Windows

書式 : -s | --copy-linkfile

=====

シンボリックリンクが参照するファイルをコピーします。このオプションを使用しない場合は、転送先でシンボリックリンクの作成を行います。

--

例 :

```
[user@localhost ~]$ hcp -s ...
```

--

1.3.8 S, follow-linkdir

=====

対応 OS : Linux / Windows

書式 : -S | --follow-linkdir

=====

シンボリックリンクが参照するディレクトリを探索します。

本オプションを使用する場合は、サイクリックなディレクトリ探索やリンクを解決しない場合に確認される転送元パス配下の情報（ファイル数や容量など）からは予想しにくい多量のデータ処理などが生じ得る点に注意が必要です。

--

例 :


```
[user@localhost ~]$ hcp -S ...  
--
```

1.3.9 e, no-emptyfile

```
=====
```

対応 OS : Linux / Windows
書式 : -e | --no-emptyfile

```
=====
```

空ファイル（サイズが 0 のファイル）のコピーを抑制します。

```
--  
例 :  
[user@localhost ~]$ hcp -e ...  
--
```

1.3.10 E, no-emptydir

```
=====
```

対応 OS : Linux / Windows
書式 : -E | --no-emptydir

```
=====
```

空ディレクトリ（配下にファイルもしくはディレクトリを全く含まないディレクトリ）のコピーを抑制します。

```
--  
例 :  
[user@localhost ~]$ hcp -E ...  
--
```

1.3.11 d, no-dotfile

```
=====
```

対応 OS : Linux / Windows
書式 : -D | --no-dotfile

```
=====
```

ドット(.)から始まるファイルのコピーを抑制します。

```
--  
例 :  
[user@localhost ~]$ hcp -d ...  
--
```

1.3.12 D, no-dotdir

```
=====
```

対応 OS : Linux / Windows

書式 : -D | --no-dotdir

=====

ドット(.)から始まるディレクトリのコピーを抑制します。

--

例 :

```
[user@localhost ~]$ hcp -D ...
```

--

1.3.13 l, copy-hidden

=====

対応 OS : Windows

書式 : -I | --copy-hidden

=====

隠し属性が設定されたファイルをコピーするか指定します。

--

例 :

```
[user@localhost ~]$ hcp -I ...
```

--

1.3.14 A, archive-check

=====

対応 OS : Windows

書式 : -A | --archive-check

=====

ファイルのアーカイブ属性が設定されているファイルをコピーします。

また、転送先の OS が Windows の場合は本オプションを指定しない場合を含め、その OS の機能に従ってアーカイブ属性が適時設定されます（新規作成、更新時、など。本ソフトウェアは同属性を設定する動作は行いません）。

--

例 :

```
[user@localhost ~]$ hcp -A ...
```

--

1.3.15 r, resume

=====

対応 OS : Linux / Windows

書式 : -r <run-record-path> | --resume=<run-record-path>

run-record-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

=====
hcp コマンドの実行記録 (.hcp.out に出力されるファイル転送の実行記録) を使用して、コマンドの再開 (レジューム) を行います。

--

例 :

```
[user@localhost ~]$ hcp ...  
[user@localhost ~]$ mv .hcp.out .hcp.in  
[user@localhost ~]$ hcp -r .hcp.in ...
```

--

1.3.16 N, no-send

=====
対応 OS : Linux / Windows

書式 : -N | --no-send

=====
ネットワークへのデータ送信を抑制します。

このオプションは、転送元でのディスク I/O やデータ処理の性能を確認する目的で使用します。

--

例 :

```
[user@localhost ~]$ hcp -N ...
```

--

1.3.17 n, no-diskio

=====
対応 OS : Linux / Windows

書式 : -n <bench_spec> | --no-diskio=<bench_spec>

bench_spec := <number_of_files>:<file_size>

number_of_files

既定値 : なし

値の範囲 : 符号なし整数

file_size

既定値 : なし

値の範囲 : 符号なし倍長整数

=====
転送元及び転送先でのローカル I/O を抑制します。

このオプションは、ネットワーク通信の性能を確認する目的で使⽤します。オプションの各パラメータの意味は次の通りです。

number_of_files := 送信するファイル数
file_size := ファイル毎に送信するサイズ (単位 バイト)

オプションの値を"0:0"で指定した場合は、転送元のパスを通常の動作で探索してファイル転送を行います (**permission** オプションのためのファイル属性情報の参照やファイルからのデータの読み込みなどの処理は行われません)。

本オプションは、転送元に **/dev/zero** を指定した場合にも使⽤します。この場合は、本オプションの **file_size** で指定されたサイズだけ **/dev/zero** からデータを読み取り転送を行い、転送先では実際にファイルにデータを書き込みます (前述の動作と異なり I/O 処理は省略されません)。通常は、転送先に **/dev/null** を指定して、スペシャルデバイスでの読み書きのみを追加で行い、通信パフォーマンスを確認する目的で使⽤します。

```
--
例 :
[user@localhost ~]$ hcp -n 1000:1048576 ...
例 2:
[user@localhost ~]$ hcp -n 1:1048576 /dev/zero 192.168.10.100:874:/dev/null
--
```

1.3.18 hpfp

```
=====
対応 OS : Linux / Windows
書式 : --hpfp
=====
```

トランスポートに HpFP プロトコルを使用するように指示します。

```
--
例 :
[user@localhost ~]$ hcp ... --hpfp /path/to/src_file 192.168.10.100:65520:/path/to/dst_file
--
```

1.3.19 wss

```
=====
対応 OS : Linux / Windows
書式 : --wss
=====
```

トランスポートに WebSocket プロトコル(SSL/TLS)を使用するように指示します。

1.3.20 V, version

=====
対応 OS : Linux / Windows

書式 : -V | --version
=====

hcp コマンドのバージョンを表示します。

--

例 :

```
[user@localhost ~]$ hcp -V
```

```
hcp client (hcp) 1.3.0_42 / Linux (HpFP2 2.0.0.91_16)
```

--

1.3.21 config-test

=====
対応 OS : Linux / Windows

書式 : --config-test
=====

hcp コマンドの入力パラメータ及び設定情報を出力します。

--

例 :

```
[user@localhost ~]$ hcp -t
```

...

```
Number of Logical Processors : 2
```

```
Number of Physical Processors : 2
```

Command parameters

```
permission      : disable
```

```
recursive       : disable
```

```
anydirs         : disable
```

```
regex           : disable
```

```
verify          : disable
```

```
compress        : disable
```

```
copy-linkfile   : disable(don't copy symlink file)
```

```
follow-linkdir  : disable(don't follow symlink directory)
```

```
no-emptyfile    : disable(copy)
```

```
no-emptydir     : disable(copy)
```

```
no-dotfile      : disable(copy)
```

```
no-dotdir       : disable(copy)
```

```
copy-hidden     : disable(don't copy)
```

```
archive-check : disable(don't check archive property)
resume        : disable
no-send       : disable
no-diskio     : disable (0, 0)
copy-mode     : disable [ALLCOPY]
overwrite     : disable [FORCE]
fail-action   : disable [HALT]
source-file   : disable
source-file-host : disable
port          : disable
overwrite-als-temp-file : disable
preserve-win-read-only : disable
auto-resume   : disable
no-out-of-order-header-messaging : disable
reading-dev   : disable
reading-dev-via-mmap : disable
transfer-expire : 0 sec
hpfp          : disable (TCP used)
ws            : disable (TCP used)
ws-proxy      : -
wss-no-check-certificate : disable
mcd           : -
version       : disable
config-test   : enable
help          : disable
user          : disable
password      : disable
config-file   : disable
log-file      : disable
stat-log-file : disable
hcp-out       : - [.hcp.out]
multi-run     : disable
```

Target source:

Target destination:

Configuration parameters

```
PubkeyAuthentication : yes
WinLogonUserAuthentication : yes
PAMAuthentication : yes
LocalPasswordAuthentication : yes
RequireServerCertificateSecurity : yes
RejectFallbackServerCertificateSecurity : yes
IgnoreCertificateCNInvalid : no
IgnoreCertificateDateInvalid : no
```

```

IgnoreUnknownCA           : no
IgnoreRevocation          : no
WSSIgnoreCertificateCNInvalid : no
WSSIgnoreCertificateDateInvalid : no
WSSIgnoreUnknownCA       : no
WSSIgnoreRevocation       : no
WSSOptions                : -
WSSCipherSuites           : -
WSSCipherList             : -
CompressLevel             : -1
HeaderCompress            : yes
ContentCompress           : yes
OCSPRevocationEnabled     : yes
StrictHostKeyChecking     : ask
ApplicationStatLog        : yes Rotation[size=no:0:0, pattern=no:]
TransportStatLog          : no Rotation[size=no:0:0, pattern=no:]
PrivateKeySearchDir       : - [/etc/hcp/keys]
PrivateKeyFile            : - [~/etc/hcp/id_rsa]
CACertificateFile         : - [/etc/hcp/cacert.pem]
CACertificatePath         : -
CARevocationFile         : - [/etc/hcp/crl.pem]
CARevocationPath         : -
WSSCACertificateFile      : -
ProtocolVersion           : 2
MaxConcurrentThread       : 0
UDPTransportExtensionBufferSize : 2147483648
TCPTransportSocketSendBuffer : 0
MaxRequestFileEntryAtOnce : 50
MaxReceiveFileSize       : unlimited
MaxSendFileSize          : unlimited
MaxBufferSize            : 1073741824
MaxReceiveRate            : 10000000000
MaxSendRate              : 10000000000
InitHeaderBlockSize      : 51200
InitContentBlockSize     : 1048576
MaxHeaderBlockSize       : 51200
MaxContentBlockSize      : 1048576
TransportTimeout         : 180
FileLock                 : no
FileLockTrials            : 0
FileLockTrialInterval    : 3
AcceptableCryptMethod     : AES256/CTR/VMAC AES256/CBC AES128/CBC
[Intel:AES-NI=yes]
AcceptableDigestMethod    : MM128 SHA256 SHA160
DisableDataIntegrityChecking : no
AcceptDataIntegrityCheckingOnRejection : yes

```

```
TransportCharEncoding      : UTF8
HostEncoding               : UTF8
ApplicationLog              : INFO no Rotation[size=no:0:0, pattern
=no:]
PubkeyAuthenticationPrior  : yes
ApplicationStatLogSecurityEx : yes
UseProperCopyAndSync       : yes
AtomicLikeSaving           : no .tmp NONE [threshold=0]
AutoResumeTrials           : unlimited (interval 3 secs)
MemoryTransferConcurrency  : 2 (wait_type cond, exp_nsec 1)
MaxReadRate                : unlimited
MaxWriteRate               : unlimited
--
```

1.3.22 h, help

```
=====
対応 OS : Linux / Windows
書式 : -h | --help
=====
```

hcp コマンドのヘルプを表示します。

```
--
例 :
[user@localhost ~]$ hcp -h
--
```

1.3.23 m, copy-mode

```
=====
対応 OS : Linux / Windows
書式 : -m <mode_name> | --copy-mode=<mode_name>
-----
mode_name
既定値 : ALLCOPY
値の範囲 : ALLCOPY, UPDATE, DIFF, DIFF_STRICT, SYNC
=====
```

ファイルのコピー動作モードを指定します。

ALLCOPY は、全てのファイルをコピーします。

UPDATE は、転送先のファイルの更新日時を比較して転送元のファイルの更新日時の方が新しいファイルをコピーします。

DIFF は、UPDATE に加えてファイルのサイズ比較も行ってサイズに変化がある場合もコピーを行います。

DIFF_STRICT は、DIFF に加えてファイル毎にハッシュの比較を行い、異なるファイルをコピーします。

SYNC は、転送元に存在しない転送先のファイルを削除します（同期）。

```
--  
例 :  
[user@localhost ~]$ hcp -m UPDATE ...  
--
```

1.3.24 o, overwrite

=====
対応 OS : Linux / Windows

書式 : -o <overwrite_name> | --overwrite=<overwrite_name>

overwrite_name

既定値 : FORCE

値の範囲 : CONFIRM, FORCE, RENAME, BACKUP

=====
ファイルの上書き動作を指定します。

CONFIRM は、上書きの確認を行います。

FORCE は、確認せずに上書きを行います。

RENAME は、上書きを行う前に古いファイルを次の様な接尾辞を付加してリネームします。

.YYMMDD_HHMMSS.NNN (NNN は連番)

BACKUP は、上書きする前に古いファイルを次の様なバックアップディレクトリへ移動します。

./YYMMDD_HHMMSS/

このディレクトリ名は、トランザクション形成時にプロセスユニークに生成されます。

```
--  
例 :  
[user@localhost ~]$ hcp -o CONFIRM ...  
--
```

1.3.25 a, fail-action

=====
対応 OS : Linux / Windows

書式 : `-a <action_name> | --fail-action=<action_name>`

action_name

既定値 : `HALT`

値の範囲 : `HALT, SKIP`
=====

ファイルコピーでのエラー発生時の動作を指定します。

HALT は、ファイルコピーで継続不可能な事象が発生した場合にトランザクションを中断します。

SKIP は、ファイルのコピーでエラーが発生した場合に、同ファイルを飛ばして継続することが可能な場合は、処理を続けます。

--

例 :

`[user@localhost ~]$ hcp -a SKIP ...`

--

1.3.26 user

=====

対応 OS : `Linux / Windows`

書式 : `--user=<username>`

username

既定値 : なし

値の範囲 : ユーザ名文字列
=====

ユーザ認証で使用するユーザ名を指定します。

--

例 :

`[user@localhost ~]$ hcp --user=user ...`

--

1.3.27 password

=====

対応 OS : `Linux / Windows`

書式 : `--password=<password>`

password

既定値 : なし

値の範囲 : パスワード文字列
=====

ユーザ認証で使用する資格情報（パスワード、パスフレーズ）を指定します。

```
--
例 :
[user@localhost ~]$ hcp -password=password ...
--
```

1.3.28 f, source-file

```
=====
対応 OS : Linux / Windows
書式 : -f <source-path-list-file> | --source-file=<source-path-list-file>
-----
source-path-list-file
既定値 : なし
値の範囲 : ファイルシステムのパス文字列
=====
```

転送元の一覧を記述したファイルを指定します。

```
--
例 1: ローカルが転送元

[user@localhost ~]$ cat source.list
file1.txt
child_dir/file2.txt
child_dir2/
/home/user/file3.txt

[user@localhost ~]$ hcp -f source.list ...
--
例 2: リモートが転送元 1
```

```
[user@localhost ~]$ cat source.list
192.168.100.100:874:file1.txt
:child_dir/file2.txt
:child_dir2/
:/home/user/file3.txt

[user@localhost ~]$ hcp -f source.list ...
--
例 3: リモートが転送元 2
```

```
[user@localhost ~]$ cat source.list
file1.txt
child_dir/file2.txt
child_dir2/
```

/home/user/file3.txt

```
[user@localhost ~]$ hcp -f source.list --host=127.0.0.1 ...
```

--

例 4: リモートが転送元 3

```
[user@localhost ~]$ cat source.list
192.168.100.100:874:file1.txt
192.168.100.100:874:child_dir/file2.txt
192.168.100.100:874:child_dir2/
192.168.100.100:874:/home/user/file3.txt
```

```
[user@localhost ~]$ hcp -f source.list ...
```

--

1.3.29 --source-file-host

=====

対応 OS : Linux / Windows

書式 : --source-file-host=<remote-host>[:<remote-port>]

remote-host

既定値 : なし

値の範囲 : IP アドレス もしくは ホスト名

remote-port

既定値 : なし

値の範囲 : ポート番号

=====

source-file オプションを指定する際に使用する接続先のリモートホストを指定します。

--

例 :

```
[user@localhost ~]$ hcp --source-file-host=192.168.100.100 ...
```

--

1.3.30 port

=====

対応 OS : Linux / Windows

書式 : --port=<remote-port>

remote-port

既定値 : なし

値の範囲 : ポート番号

接続先のリモートホストのサービスポート番号を指定します。—source-file-host
を使用する場合は、こちらのポート番号指定が優先されます。

--

例 :

```
[user@localhost ~]$ hcp --port=1874 ...
```

--

1.3.31 config-file

対応 OS : Linux / Windows

書式 : --config-file=<config-file-path>

config-file-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

hcp コマンドが使用する設定ファイルのパスを指定します。

hcp コマンドは、次の順に設定ファイルの読み込みを行います。

1. /etc/hcp/hcp.conf
2. <ユーザホームディレクトリ>/hcp/hcp.conf
3. <本オプションで指定されたパス>

ファイルが存在しない場合は、読み込みはスキップされます。いずれのファイルでも設定ファイルの読み込みが成功しなかった場合は、設定ファイルの読み込みエラーとなり動作は停止します。

--

例 :

```
[user@localhost ~]$ hcp --config-file=hcp.conf ...
```

--

1.3.32 log-file

対応 OS : Linux / Windows

書式 : --log-file=<log-file-path>

log-file-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

hcp コマンドのログを指定したパスのファイルに出力します。

--

例 :

```
[user@localhost ~]$ hcp --log-file=hcp.log ...
```

--

1.3.33 stat-log-file

対応 OS : Linux / Windows

書式 : --stat-log-file=<log-file-path>

log-file-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

hcp コマンドが出力する統計ログの基準パスを指定します。

指定されたパスに接尾辞を付加して各統計ログが出力されます。

<指定されたパス>.application (アプリケーション統計)

<指定されたパス>.transport.tcp (TCP トランスポート統計)

<指定されたパス>.transport.hpfp (HpFP トランスポート統計)

<指定されたパス>.transport.ws (WS/WSS トランスポート統計)

--

例 :

```
[user@localhost ~]$ hcp --stat-log-file=.hcp.statistics2 ...
```

--

1.3.34 hcp-out

対応 OS : Linux / Windows

書式 : --hcp-out=<output-path>

output-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

ファイル転送の実行記録を出力するファイルを指定します。本オプションで出力される情報は、resume オプションで必要となります。

--

例：

```
[user@localhost ~]$ hcp -v ...
SRC /home/user/Desktop/hcp_src5
DST 127.0.0.1:11112:/home/user/Desktop/hcp_dst5
OK 0000 FS 80000001 /home/user/Desktop/hcp_src5/file1.txt
OK 0000 FT 00000001 /home/user/Desktop/hcp_src5/file1.txt
EXIT 0 REASON 0000
```

--

本オプションを指定しない場合は、実行記録は実行ディレクトリの次のファイルなどに記録されます。

.hcp.out (Linux)
_hcp.out (Windows)

“-”を指定した場合は、標準出力に出力します。

FT (File Transfer)は、ファイル転送処理を表します。書式は次の通りです。

<result> <reason> FT <sequence> [<src_label>]<path>

result は、処理結果(OK もしくは NG)が記録されます。

reason は、その処理結果の理由として後述のファイル処理理由コードが記録されます。

sequence は、この処理に割り当てられた番号（シーケンス番号）が記録されます。

src_label は、この処理で扱われるファイルに対応するソースのラベルが次の書式で記録されます。

SRC<src_index>

src_index は、コマンドで指定したソースの添え字が記録されます。

path は、この処理で扱われるファイルパスが記録されます。コマンドを実行しているホストのパスが記録されます。

複数のソースが指定された場合は、次の様な内容で進捗情報が記録されます。

—

例：

```
SRC0 /home/user/Desktop/hcp_src5
SRC1 /home/user/Desktop/hcp_src6
DST 127.0.0.1:11112:/home/user/Desktop/hcp_dst5
OK 0000 FS 80000001 SRC0 /home/user/Desktop/hcp_src5/file1.txt
```

```
OK 0000 FS 80000002 SRC0 /home/user/Desktop/hcp_src5/file2.txt
OK 0000 FS 80000003 SRC1 /home/user/Desktop/hcp_src6/file3.txt
OK 0000 FS 80000004 SRC1 /home/user/Desktop/hcp_src6/file4.txt
OK 0000 FT 00000001 SRC0 /home/user/Desktop/hcp_src5/file1.txt
OK 0000 FT 00000002 SRC0 /home/user/Desktop/hcp_src5/file2.txt
OK 0000 FT 00000003 SRC1 /home/user/Desktop/hcp_src6/file3.txt
OK 0000 FT 00000004 SRC1 /home/user/Desktop/hcp_src6/file4.txt
EXIT 0 REASON 0000
```

—

FS (File Sync)は、ファイル同期処理を表します。転送先のファイルが転送元に存在せず本同期処理によって削除される場合は、次の様に記録されます。

—

例：

```
SRC /home/user/Desktop/hcp_src5
DST 127.0.0.1:11112:/home/user/Desktop/hcp_dst5
OK 0000 FS 80000001 /home/user/Desktop/hcp_src5/file1.txt
OK 0000 FS 80000002 /home/user/Desktop/hcp_src5/file2.txt
OK A001 FS 80000003 /home/user/Desktop/hcp_src5/file3.txt
OK 0000 FT 00000001 /home/user/Desktop/hcp_src5/file1.txt
OK 0000 FT 00000002 /home/user/Desktop/hcp_src5/file2.txt
EXIT 0 REASON 0000
```

—

ソースが複数ある場合は、ソースラベルの添え字が"?"で表記されます。

—

例：

```
SRC0 /home/user/Desktop/hcp_src5
SRC1 /home/user/Desktop/hcp_src6
DST 127.0.0.1:11112:/home/user/Desktop/hcp_dst5
OK 0000 FS 80000001 SRC0 /home/user/Desktop/hcp_src5/file1.txt
OK 0000 FS 80000002 SRC0 /home/user/Desktop/hcp_src5/file2.txt
OK 0000 FS 80000003 SRC1 /home/user/Desktop/hcp_src6/file3.txt
OK 0000 FS 80000004 SRC1 /home/user/Desktop/hcp_src6/file4.txt
OK A001 FS 80000005 SRC? /home/user/Desktop/hcp_src6/file5.txt
OK 0000 FT 00000001 SRC0 /home/user/Desktop/hcp_src5/file1.txt
OK 0000 FT 00000002 SRC0 /home/user/Desktop/hcp_src5/file2.txt
OK 0000 FT 00000003 SRC1 /home/user/Desktop/hcp_src6/file3.txt
OK 0000 FT 00000004 SRC1 /home/user/Desktop/hcp_src6/file4.txt
EXIT 0 REASON 0000
```


最後の行にこの実行の終了ステータスと理由コードが次の書式で記録されます。

EXIT <exit_status> REASON <reason_code>

後述の設定項目 UseProperCopyAndSync を yes に設定した場合は、ディレクトリ探索時に上書きに関する判定結果が次の書式で記録されます。

<result> <reason> DE <sequence> [<src_label>]<path>

1.3.35 multi-run

対応 OS : Linux / Windows

書式 : --multi-run=<record-path>[:<output-switch>]

record-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

output-switch

書式 : ((A|T|R|L)[...] | THRU | FULL)

既定値 : THRU

クライアントを多重起動モードで起動します。本オプションを指定した場合は、その指定に従い次の記録（ログ）を指定のディレクトリにユニークなファイル名で保存します。同一のディレクトリで複数のクライアントで作業する場合や、バックグラウンド実行を行う場合などに利用します。

- アプリケーション統計
- トランスポート統計
- 結果出力
- アプリケーションログ

record-path は、前述の記録（ログ）を保存するディレクトリを指定します。

output-switch は、前述の記録（ログ）の出力を制御する指示を指定します。

A はアプリケーション統計を表し、指定すると上記ディレクトリに次のファイル名でアプリケーション統計を記録します。

hcp.mr.<YYYYMMDD_hhmmss_msec>.<pid>.statistics.application

T はトランスポート統計を表し、指定すると上記ディレクトリに次のファイル名でトランスポート統計を記録します。

`hcp.mr.<YYYYMMDD_hhmmss_msec>.<pid>.statistics.application`

R は実行記録を表し、指定する上記ディレクトリに次のファイル名でコマンドの実行記録を記録します。

`hcp.mr.<YYYYMMDD_hhmmss_msec>.<pid>.out`

L はアプリケーションログを表し、指定すると上記ディレクトリに次のファイル名でログを記録します。

`hcp.mr.<YYYYMMDD_hhmmss_msec>.<pid>.log`

FULL は、“ATRL”を指定した場合と同じ動作をします。

THRU は、設定ファイルやコマンド引数の指定に従って記録の出力を行います。設定ファイルでアプリケーション統計（トランスポート統計）が有効に設定されている場合は、上記のファイル名にアプリケーション統計（トランスポート統計）を記録します。結果出力（アプリケーションログ）がファイル出力に設定されている場合（-v オプション未指定、-l オプション指定）は、上記のファイル名に実行記録（ログ）を記録します。

--

例：

`[user@localhost ~]$ hcp --multi-run=/var/tmp:ATR // ログは標準出力に出す。`
他は、/var/tmp 配下に出力

--

1.3.36 overwrite-als-temp-file

=====

対応 OS : Linux / Windows

書式 : `--overwrite-als-temp-file`

=====

アトミックライクな保存で既に存在する一時ファイルを上書きするように要求します。

このオプションを指定せずに一時ファイルが既に存在する場合は、そのファイルの転送はエラーと判定され中断されます。

また、サーバの設定 `AtomicLikeSavingRejectOverwriteRequest` が `yes` の場合は、このオプションが指定されていてもサーバ上で既に存在する一時ファイルの上書きは拒否されます。

1.3.37 preserve-win-read-only

```
=====
対応 OS : Linux / Windows
書式 : --preserve-win-read-only
=====
```

転送先で Windows の読み取り専用属性を保持します。転送元で Windows の読み取り専用属性が有効な場合や、Unix の所有者のアクセス権が読み込みのみの場合に適用されます。

1.3.38 udp

```
=====
対応 OS : Linux / Windows
書式 : --udp=<hpfp_options>
  hpfp_options := <hpfp_udp_port>:<hpfp_cong_mode>:<hpfp_sndbuf>:<hpfp_rcvbuf>:<hpfp_mss>
-----
```

hpfp_udp_port

書式 : (DEFAULT | D | <decimal_number>)

既定値 : 65520

値の範囲 : decimal_number は、1 - 65535 の範囲。D は DEFAULT の略記。

hpfp_cong_mode

既定値 : FAIR

値の範囲 : DEFAULT(D), FAIR(F), MODEST(M), FAIR_FAST_START(S), AGGRESSIVE(A)

※括弧内は省略表記

hpfp_sndbuf

書式 : (DEFAULT | D | <decimal_number>[[(T|G|M|K)]B])

既定値 :

100MB (Linux.x86 / Windows)

8MB (Raspbian)

値の範囲 : バイト換算で符号なし倍長整数の範囲。D は DEFAULT の略記。

hpfp_rcvbuf

書式 : (DEFAULT | D | <decimal_number>[[(T|G|M|K)]B])

既定値 :

200MB (Linux.x86 / Windows)

16MB (Raspbian)

値の範囲 : バイト換算で符号なし倍長整数の範囲。D は DEFAULT の略記。

hpfp_mss

書式 : (DEFAULT | D | NONE | N | <decimal_number>[[(T|G|M|K)]B])

既定値 : NONE

値の範囲 : バイト換算で符号なし整数の範囲。D は DEFAULT の略記。N は NONE の略記。

=====

本オプションは、廃止予定(deprecated)です。

トランスポートに HpFP プロトコルを使用するように指示します。

hpfp_udp_port は、HpFP プロトコルで使用する UDP ポート番号を指定します。

hpfp_cong_mode は、輻輳制御モードを指定します。

hpfp_sndbuf は、HpFP プロトコル送信バッファサイズを指定します (バイト単位)。

hpfp_rcvbuf は、HpFP プロトコル受信バッファサイズを指定します (バイト単位)。

hpfp_mss は、HpFP プロトコルの MSS(Maximum Segment Size)を指定します。MTU サイズからプロトコルヘッダ 44 バイトを引いたサイズが最適値です。D を指定すると既定値を使用します。N を指定した場合、HpFP プロトコルによる MTU 探索を行い値を決定します。

輻輳制御モードについては、指定したモードをサーバが許可しない場合は、FAIR が適用されます。この動作により異なるアルゴリズムが適用される場合は、次のようなログにより通知されます。

```
2018/07/05 13:20:44 00007f9a2dfa0700:INFO :Your requested congestion mode
of HpFP will not be used (requested=AGGRESSIVE, actual=FAIR).
```

--

例 :

```
[user@localhost ~]$ hcp -U D:S:4MB:8MB:8956B ...
```

--

1.3.39 ws

=====

対応 OS : Linux / Windows

書式 : --ws

=====

トランスポートに WebSocket プロトコル (平文通信) を使用するように指示します。

1.3.40 wss-no-check-certificate

```
=====
対応 OS : Linux / Windows
書式 : --wss-no-check-certificate
=====
```

WebSocket プロトコルで SSL/TLS を使用する際に、サーバ証明書の検証を無効にするように指示します。証明書の CN(Common Name)名、期間、中間証明書・ルート証明書の検証を省略してサーバ証明書を受け付けます。

1.3.41 ws-proxy

```
=====
対応 OS : Linux / Windows
書式 : --ws-proxy=<proxy_server>
-----
```

```
proxy_server
書式 : <proxy_server_address>:<proxy_server_port_number>
既定値 : なし
値の範囲 : プロキシサーバのアドレスとポート番号
=====
```

WebSocket プロトコルを使用する場合に使用するプロキシサーバを指定します。SSL/TLS を使用する場合、使用しない場合ともに適用されます。指定しない場合は、システムのプロキシ設定を参照して動作します。

1.3.42 hpfp-cong

```
=====
対応 OS : Linux / Windows
書式 : --hpfp-cong=<hpfp_cong_mode>
-----
```

```
hpfp_cong_mode
既定値 : FAIR
値の範囲 : FAIR(F), MODEST(M), FAIR_FAST_START(S), AGGRESSIVE(A)
※括弧内は省略表記
=====
```

HpFP プロトコルで使用する輻輳制御モードを指定します。

輻輳制御モードについては、指定したモードをサーバが許可しない場合は、FAIR が適用されます。この動作により異なるアルゴリズムが適用される場合は、次のようなログにより通知されます。

```
2018/07/05 13:20:44 00007f9a2dfa0700:INFO :Your requested congestion mode
of HpFP will not be used (requested=AGGRESSIVE, actual=FAIR).
```

--

例 :

```
[user@localhost ~]$ hcp ... --hfp --hfp-cong=S /path/to/src_file 192.168.10.100:65520:/path/to/dst_file
```

--

1.3.43 hfp-mss

=====

対応 OS : Linux / Windows

書式 : --hfp-mss=<hfp_mss>

hfp_mss

書式 : (NONE | N | <decimal_number>[[(T|G|M|K)]B])

既定値 : NONE

値の範囲 : バイト換算で符号なし整数の範囲。N は NONE の略記。

=====

HpFP プロトコルの MSS(Maximum Segment Size)を指定します。MTU サイズからプロトコルヘッダ 44 バイトを引いたサイズが最適値です。N を指定した場合、HpFP プロトコルによる MTU 探索を行い値を決定します。

--

例 :

```
[user@localhost ~]$ hcp ... --hfp --hfp-mss=8956 /path/to/src_file 192.168.10.100:65520:/path/to/dst_file
```

--

1.3.44 hfp-sndbuf

=====

対応 OS : Linux / Windows

書式 : --hfp-sndbuf=<hfp_sndbuf>

hfp_sndbuf

書式 : <decimal_number>[[(T|G|M|K)]B]

既定値 :

100MB (Linux.x86 / Windows)

8MB (Raspbian)

値の範囲 : バイト換算で符号なし倍長整数の範囲。

=====

HpFP プロトコルの送信バッファサイズを指定します (バイト単位) 。

--

例 :

```
[user@localhost ~]$ hcp ... --hfp --hfp-sndbuf=8MB /path/to/src_file 19
```

2.168.10.100:65520:/path/to/dst_file

--

1.3.45 hpfp-rcvbuf

対応 OS : Linux / Windows

書式 : --hpfp-rcvbuf=<hpfp_rcvbuf>

hpfp_rcvbuf

書式 : <decimal_number>[[(T|G|M|K)]B]

既定値 :

200MB (Linux.x86 / Windows)

16MB (Raspbian)

値の範囲 : バイト換算で符号なし倍長整数の範囲。

HpFP プロトコルの受信バッファサイズを指定します (バイト単位) 。

--

例 :

[user@localhost ~]\$ hcp ... --hpfp --hpfp-rcvbuf=16MB /path/to/src_file 1
92.168.10.100:65520:/path/to/dst_file

--

1.3.46 reading-dev

対応 OS : Linux / Windows

書式 : --reading-dev

転送元にブロックデバイスもしくはキャラクタデバイスを指定した場合に、read
関数を用いてデータを読み込みます。

1.3.47 reading-dev-via-mmap

対応 OS : Linux / Windows

書式 : --reading-dev-via-mmap

転送元にブロックデバイスもしくはキャラクタデバイスを指定した場合に、
Mapped I/O を用いてデータを読み込みます。

1.3.48 transfer-expire

対応 OS : Linux / Windows

書式 : --transfer-expire=<expire_sec>

expire_sec

既定値 : なし

値の範囲 : 期限の秒数

ファイル転送を指定した秒数で中断します。ファイル転送が指定した秒数内で終了する場合は、中断せず完了します。

1.3.49 no-out-of-order-header-messaging

対応 OS : Linux / Windows

書式 : `--no-out-of-order-header-messaging`

制御メッセージ（ファイル要求などを含むメッセージ）のアウトオブオーダーを抑制します。多重接続モード時にプライマリ接続で制御メッセージの送信のみが行われ、セカンダリ接続でコンテンツデータ（ファイルのペイロードを含むメッセージ）の送信のみが行われる動作を解除します。全ての接続で全てのメッセージが送信されるように動作が変わります。

1.3.50 mcd

対応 OS : Linux / Windows

書式 : `--mcd=<multiple_connection_degree>`**multiple_connection_degree**

既定値 : なし

値の範囲 : 1 - 65535

多重接続を行う場合の接続数を指定します。指定しない場合は、サーバのサービスに設定されている多重接続数の上限まで接続を作成します。1 を指定した場合は、単一接続での通信動作を行います。また、サーバのサービスに設定されている多重接続数の上限を超える値を指定した場合は、この上限まで接続を作成します。

1.3.51 auto-resume

対応 OS : Linux / Windows

書式 : `--auto-resume`

ネットワークの障害で中断した場合にコピー処理を自動で再開するように指示します。Ctrl+Cなどでユーザから中断を指示した場合は、その時点で再開はせず停止します。

パスワードで暗号化されていない秘密鍵を使用した公開鍵認証で利用を推奨します。パスワードが必要な場合、再認証による対話動作を回避するために認証情報をメモリ上にキャッシュします。

1.4 hsync (実験的)

Local : hsync [OPTION]... SRC [SRC]... DEST

Remote :

Push : hsync [OPTION]... SRC [SRC]... [USER@]HOST[:PORT]:DEST

Pull : hsync [OPTION]... [USER@]HOST[:PORT]:SRC [:SRC]... DEST

hsync コマンドは、リモート（サーバ、hcpd デーモン）との間でファイル同期を行うコマンドです。ローカルのファイル同期機能もサポートします。

短縮名	オプション名	概略
v	verbose	ログの冗長レベルを上げる
	info	INFO レベルのログの冗長レベルを指定する (予約)
	debug	DEBUG レベルのログの冗長レベルを指定する (予約)
q	quiet	ログを抑制
c	checksum	ダイジェスト検査に基づくファイル送信判定
a	archive	アーカイブモード (rlptgoD を指定)
r	recursive	ディレクトリの再帰探索
R	relative	相対パス名を使用する (予約)
	no-implied-dirs	
b	backup	バックアップ
	backup-dir	バックアップディレクトリ指定
	suffix	バックアップ接尾辞指定
u	update	新しいファイルをコピーする
	inplace	インプレース式ファイルデータコピー (予約)
d	dirs	再帰探索せずにディレクトリをコピーする
l	links	シンボリックリンクをそのままコピーする

L	copy-links	シンボリックリンクの参照先の実体をコピーする
	copy-unsafe-links	“unsafe”なファイルは実体をコピーする
	safe-links	“unsafe”なシンボリックリンクはそのままコピーしない
k	copy-dirlinks	ディレクトリを参照するシンボリックリンクは実体をコピーする
K	keep-dirlinks	送信先のシンボリックリンクされたディレクトリをリンクではなくディレクトリとして扱う（予約）
p	perms	パーミッションを保持する
E	executability	実行性を保持する
	chmod	指定したパーミッションを適用する
o	owner	所有者を保持する
g	group	グループを保持する
	devices	デバイスファイルを保持する（予約）
	specials	スペシャルファイルを保持する（予約）
D		devices と specials を指定（予約）
t	times	更新日時を保持する
O	omit-dir-times	更新日時の保持からディレクトリを除外
J	omit-link-times	更新日時の保持からシンボリックリンクを除外
	super	受信側で特権ユーザ動作を試行する（予約）
S	sparse	スパースファイルのサポート（予約）
n	dry-run	ファイル进行操作せずに試行する
W	whole-file	ファイル全体をコピーする
	existing	存在するファイルのみコピー
	ignore-existing	存在するファイルはコピーしない
	del	delete-during のエイリアス
	delete	転送元に存在しない転送先ファイルを削除する
	delete-before	削除を事前に実行する（予約）
	delete-during	転送中に削除も実行する
	delete-delay	転送中に削除対象をチェックして転送後に削除を行う（予約）
	delete-after	転送後に削除を実行する（予約）

	delete-excluded	除外されたファイルを転送先から削除する（予約）
	ignore-errors	I/O エラーを無視して削除を実行する（予約）
	force	非空ディレクトリの削除を実行する
	max-delete	削除ファイル数の上限設定
	max-size	送信するファイルサイズの上限設定
	min-size	送信するファイルサイズの下限設定
	partial	転送中途ファイルの保持（予約）
	partial-dir	転送中途ファイルを指定のディレクトリに保管する（予約）
	delay-updates	更新するファイルは最後に置き換える（予約）
m	prune-empty-dirs	空ディレクトリコピーの抑制
	numeric-ids	ユーザ名グループ名ではなく UID/GID 値を適用する
	usermap	ユーザ名マッピング
	groupmap	グループ名マッピング
	chown	簡易ユーザ名グループ名マッピング
I	ignore-times	更新日時とサイズによるスキップを無効化
	size-only	サイズが同じファイルはスキップする
@	modify-window	更新日時の比較ウィンドウの設定（予約）
T	temp-dir	一時ディレクトリを作成して転送する（予約）
z	compress	転送中にファイルデータの圧縮を行う
	compress-level	圧縮レベルの指定
f	filter	フィルタルールの追加
F		規定のフィルタルールの追加（予約）
	exclude	除外ルールの追加
	exclude-from	除外ルールをファイルから読み込む
	include	含有ルールの追加
	include-from	含有ルールをファイルから読み込む
	files-from	転送元ファイルのリストをファイルから読み込む（予約）
0	from0	ルールとファイルリストの終端コードの変更（予約）

s	protect-args	引数展開の保護（予約）
	stats	ファイル転送の統計情報の追加（予約）
8	8-bit-output	文字列のハイビットエスケープ設定（予約）
h	human-readable	数値をヒューマンリーダブルに表示する（予約）
	progress	転送中に経過情報を表示する（予約）
P		partial と progress 指定（予約）
i	itemize-changes	ファイルの更新で変更のサマリを表示する（予約）
	out-format	指定した書式で更新の表示を行う（予約）
	log-file	ログの出力先の指定
	bwlimit	通信帯域を制限する
	stop-at	指定した日時で中断する（予約）
	stop-after	指定した時間で中断する
	hpfp	HpFP プロトコルを使用する
	wss	WebSocket プロトコル(SSL/TLS)を使用する
V	version	バージョンを表示する
	config-test	入力パラメータ及び設定ファイルのロード結果を表示する
h	help	コマンドラインヘルプを表示する
	user	ユーザ名
	password	パスワード
	config-file	設定ファイル
	hcp-log-file	HCP ログファイル
	hcp-stat-log-file	HCP 統計ログファイル
	udp	UDP (HpFP)プロトコルを使用する（ポート分離型）
	ws	WebSocket プロトコル（平文通信）を使用する
	wss-no-check-certificate	WebSocket SSL/TLS サーバ証明書検証無効化
	ws-proxy	WebSocket プロキシサーバ指定
	hpfp-cong	HpFP プロトコルの輻輳制御モードを指定
	hpfp-mss	HpFP プロトコルの MSS 指定
	hpfp-sndbuf	HpFP プロトコルの送信バッファサイズ指定

	hpfp-rcvbuf	HpFP プロトコルの送信バッファサイズ指定
	hcp-out	実行記録を出力するファイルを指定する
	investigation	調査モード
	multi-run	多重起動モード

次のオプションは hcpd コマンドの説明をご参照ください。

--investigation

次のオプションは hcp コマンドの説明をご参照ください。

--hpfp, --wss, --user, --password, --config-file, --log-file, --stat-log-file, --udp, --ws, --wss-no-check-certificate, --ws-proxy, --hpfp-cong, -hpfp-mss, --hpfp-sndbuf, --hpfp-rcvbuf, --hcp-out, --multi-run

1.4.1 v, verbose

```
=====
対応 OS : Linux / Windows
書式 : -v | --verbose
=====
```

ログの冗長性を 1 段階上げます。

本版では 2 段階以降(-vv)については該当するログはありません。

```
--
例 :
[user@localhost ~]$ hsync -v ...
[user@localhost ~]$ hsync -vv ...
--
```

1.4.2 q, quiet

```
=====
対応 OS : Linux / Windows
書式 : -q | --quiet
=====
```

ログの出力を抑制します。

本版ではすべての同期機能のログを非表示にします（今後調整の予定あり）。

```
--
例 :
[user@localhost ~]$ hsync -q ...
--
```

1.4.3 c, checksum

```
=====
対応 OS : Linux / Windows
書式 : -c | --checksum
=====
```

チェックサムが一致しない場合にファイルを送信するように動作を変更します。

```
--
例 :
[user@localhost ~]$ hsync -c ...
--
```

1.4.4 a, archive

```
=====
対応 OS : Linux / Windows
書式 : -a | --archive
=====
```

-rflptgo を指定した場合と同じ動作をします。簡単にファイルの属性などを保持しつつディレクトリをアーカイブしたい場合などに指定します。

```
--
例 :
[user@localhost ~]$ hsync -a ...
--
```

1.4.5 r, recursive

```
=====
対応 OS : Linux / Windows
書式 : -r | --recursive
=====
```

ディレクトリを再帰的に探索してファイルの送信を行います。

```
--
例 :
[user@localhost ~]$ hsync -r ...
--
```

1.4.6 b, backup

```
=====
対応 OS : Linux / Windows
書式 : -b | --backup
=====
```

転送先にファイルが存在する場合は、そのファイルをリネームしてバックアップを行います。

--

例 :

```
[user@localhost ~]$ hsync -b ...
```

--

1.4.7 backup-dir

=====
対応 OS : Linux / Windows

書式 : --backup-dir=<dir-name>

dir-name

既定値 : なし

値の範囲 : パスコンポーネント文字列
=====

バックアップするディレクトリ名を指定します。

--

例 :

```
[user@localhost ~]$ hsync --backup-dir=backup_20211013 ...
```

--

1.4.8 suffix

=====
対応 OS : Linux / Windows

書式 : --suffix=<suffix-name>
=====

バックアップ時にリネームする接尾辞を指定します。

--

例 :

```
[user@localhost ~]$ hsync --suffix=.bk ...
```

--

1.4.9 u, update

=====
対応 OS : Linux / Windows

書式 : -u | --update
=====

更新日時が転送元より新しいファイルは転送しないように指示します。また、更新日時が同じ場合は、サイズが異なるファイルは転送されます。

--

例 :

```
[user@localhost ~]$ hsync -u ...
```

--

1.4.10 d, dirs

```
=====
対応 OS : Linux / Windows
書式 : -d | --dirs
=====
```

転送元で遭遇したすべてのディレクトリを含めるように指示します。—recursive
と異なりディレクトリの内容はコピーしません（転送元ディレクトリパスの末尾
に”.”や”/”が指定された場合を除く）。

--

例 :

```
[user@localhost ~]$ find from_dir/
from_dir/
from_dir/dir01
from_dir/dir02
from_dir/dir02/file02.txt
from_dir/dir03
from_dir/dir03/dir03_1
from_dir/dir03/dir03_1/file03_1.txt
from_dir/dir03/dir03_1/file03_2.txt
from_dir/file.txt
[user@localhost ~]$ hsync --dirs ... from_dir/ to_dir/
[user@localhost ~]$ find to_dir/
to_dir
to_dir/dir01
to_dir/dir02
to_dir/dir03
to_dir/file.txt
--
```

1.4.11 l, links

```
=====
対応 OS : Linux / Windows
書式 : -l | --links
=====
```

転送元のシンボリックリンクをそのまま転送先にコピーします。

—links, —copy-links などリンクのコピー動作に関わるオプションを指定しない場
合は、シンボリックリンクは転送先にはコピーされません（スキップされます）。

--

例 :

```
[user@localhost ~]$ hsync -l ...
```

--

1.4.12 L, copy-links

```
=====
対応 OS : Linux / Windows
書式 : -L | --copy-links
=====
```

転送元のシンボリックリンクを実体（ファイルもしくはディレクトリ）に解決してから転送先にコピーします。

—links, —copy-links などリンクのコピー動作に関わるオプションを指定しない場合は、シンボリックリンクは転送先にはコピーされません（スキップされます）。

--

例 :

```
[user@localhost ~]$ hsync -L ...
```

--

1.4.13 copy-unsafe-links

```
=====
対応 OS : Linux / Windows
書式 : --copy-unsafe-links
=====
```

“unsafe”な（安全でない）シンボリックリンクは実体に解決してから転送先にコピーします。

“unsafe”なシンボリックリンクは、次のようなリンクが対象となります。

- 転送元の転送ルートからのツリー配下外のパスに解決されるようなリンク
- リンク先が絶対パスが指定されているリンク

--

例 :

```
[user@localhost ~]$ hsync --copy-unsafe-links ...
```

--

1.4.14 safe-links

```
=====
対応 OS : Linux / Windows
書式 : --safe-links
=====
```

“unsafe”なシンボリックリンクは転送先へリンクとしてコピーしないように指示します。

—copy-links や—copy-unsafe-links など実体でコピーするオプションが指定されない場合は、“unsafe”なリンクについてはコピーは行われません（スキップされます）。

```
--
例：
[user@localhost ~]$ hsync --safe-links ...
--
```

1.4.15 k, copy-dirlinks

```
=====
対応 OS : Linux / Windows
書式 : copy-dirlinks
=====
```

転送元のディレクトリへのシンボリックリンクは、そのディレクトリをたどってコピーを行うように指示します。ファイルへのシンボリックリンクは解決せずにそのままリンクをコピーします。

```
--
例：
[user@localhost ~]$ hsync --copy-dirlinks ...
--
```

1.4.16 p, perms

```
=====
対応 OS : Linux / Windows
書式 : -p | --perms
=====
```

転送元のパーミッションを保持してコピーを行うように指示します。

```
--
例：
[user@localhost ~]$ hsync -p ...
--
```

1.4.17 E, executability

```
=====
対応 OS : Linux / Windows
書式 : -E | --executability
=====
```

転送元のファイルの実効性(Executability)を判定して宛先の実行パーミッションを設定するように指示します。

所有者、グループまたはその他のいずれかに実行属性が設定されている場合、そのファイルは実効性が有ると判定します。転送元と転送先でこの実行性が異なる場合に、転送先のファイルのパーミッションを次のように変更します。

- 実行属性をオフにする場合は、すべての実行属性'x'をオフに変更する
- 実行属性をオンにする場合は、読込属性'r'が設定されているフィールドの実行属性'x'をオンに変更する

--

例 :

```
[user@localhost ~]$ hsync -E ...
```

--

1.4.18 chmod

```
=====
対応 OS : Linux / Windows
書式 : --chmod=<modifier>[,<modifier>...]
-----
```

modifier

既定値 : なし

値の範囲 : パーミッション更新指定子 (chmod コマンド準拠)

```
=====
```

指定したパーミッションをファイルに適用するように指示します。

modifier には chmod コマンドに準拠した書式が使用できます。また、本コマンド固有の拡張書式については次の文書を参照してください。

<https://download.samba.org/pub/rsync/rsync.1>

--chmod=CHMOD

--

例 :

```
[user@localhost ~]$ hsync --chmod=Dg+s,ug+w,Fo-w,+X ...
```

```
[user@localhost ~]$ hsync --chmod=D2775,F664 ...
```

--

1.4.19 o, owner

```
=====
対応 OS : Linux / Windows
書式 : -o | --owner
=====
```

転送元ファイルの所有者を保持してコピーするように指示します。

—**numeric-ids** を指定しない場合、転送先のホストで（転送元ファイルの所有者属性に設定された）ユーザ名から **UID** を解決して設定されます。また、**UID** が解決できなかった場合（ユーザが存在しない場合など）は、転送元ファイルの **UID** 値が適用されます。

```
--  
例 :  
[user@localhost ~]$ hsync -o ...  
--
```

1.4.20 **g, group**

```
=====
```

対応 OS : Linux / Windows
書式 : **-g** | **--group**

```
=====
```

転送元ファイルのグループを保持してコピーするように指示します。

—**numeric-ids** を指定しない場合、転送先のホストで（転送元ファイルのグループ属性に設定された）グループ名から **GID** を解決して設定されます。また、**GID** が解決できなかった場合（グループが存在しない場合など）は、転送元ファイルの **GID** 値が適用されます。

```
--  
例 :  
[user@localhost ~]$ hsync -g ...  
--
```

1.4.21 **t, times**

```
=====
```

対応 OS : Linux / Windows
書式 : **-t** | **--times**

```
=====
```

転送元ファイルの更新日時を保持してコピーするように指示します。

```
--  
例 :  
[user@localhost ~]$ hsync -t ...  
--
```

1.4.22 **O, omit-dir-times**

```
=====
```

対応 OS : Linux / Windows

書式 : -O | --omit-dir-times

=====

ディレクトリの更新日時は保持せずにコピーするように指示します。

--

例 :

```
[user@localhost ~]$ hsync -O ...
```

--

1.4.23 J, omit-link-times

=====

対応 OS : Linux / Windows

書式 : -J | --omit-link-times

=====

シンボリックリンクの更新日時は保持せずにコピーするように指示します。

--

例 :

```
[user@localhost ~]$ hsync -J ...
```

--

1.4.24 dry-run

=====

対応 OS : Linux / Windows

書式 : --dry-run

=====

実際のファイル I/O 操作は行わずにコピー動作を実行するように指示します。

--

例 :

```
[user@localhost ~]$ hsync --dry-run ...
```

--

1.4.25 existing

=====

対応 OS : Linux / Windows

書式 : --existing

=====

存在するファイルを対象にコピーを行うように指示します。宛先で新規作成となるファイルはスキップされます。

--

例 :

```
[user@localhost ~]$ hsync --existing ...  
--
```

1.4.26 ignore-existing

```
=====
対応 OS : Linux / Windows
書式 : --ignore-existing
=====
```

存在しないファイルを対象にコピーを行うように指示します。宛先で更新となるファイルはスキップされます。

```
--
例 :
[user@localhost ~]$ hsync --ignore-existing ...
--
```

1.4.27 del

```
=====
対応 OS : Linux / Windows
書式 : --del
=====
```

—delete-during のエイリアス。

```
--
例 :
[user@localhost ~]$ hsync --del ...
--
```

1.4.28 delete

```
=====
対応 OS : Linux / Windows
書式 : --delete
=====
```

転送元に存在しない転送先のファイルやディレクトリを削除するように指示します。

```
--
例 :
[user@localhost ~]$ hsync --delete ...
--
```

1.4.29 delete-during

=====
対応 OS : Linux / Windows

書式 : --delete-during
=====

—delete の削除を、ファイルの転送中に実行するように指示します。

—delete の動作モードのデフォルトです。

--

例 :

```
[user@localhost ~]$ hsync --delete-during ...
```

--

1.4.30 force

=====
対応 OS : Linux / Windows

書式 : --force
=====

空でないディレクトリを削除するように指示します。

本オプションは、転送元がファイルやシンボリックリンクで転送先のディレクトリを削除（上書き）する場合に、そのディレクトリが空でなくても削除するように指示する場合に使用します。このケースでこのオプションを指定しなかった場合は、転送先のディレクトリは削除されず転送元のファイルの送信はスキップされます。

--

例 :

```
[user@localhost ~]$ hsync --force ...
```

--

1.4.31 max-delete

=====
対応 OS : Linux / Windows

書式 : --max-delete=<max-num>

max-num

既定値 : -1

値の範囲 : -1, 0 - 符号付き整数の最大値
=====

削除するファイルやディレクトリの上限数を指定します。

max-num には、この上限数を指定します。-1 を指定した場合は、上限なしと解釈されます。また、0 を指定した場合は、削除は行われなくなります。

```
--
例 :
[user@localhost ~]$ hsync --max-delete=100 ...
--
```

1.4.32 max-size

```
=====
対応 OS : Linux / Windows
書式 : --max-size=<size>
-----
```

```
size
既定値 : なし
値の範囲 : 数値文字列、単位付きサイズ指定文字列
=====
```

転送するファイルの最大サイズを指定します。

size には、数値文字列もしくは単位 (B, K, M, G, T, P) 付きのサイズ文字列を指定します。単位付き文字列の書式については、下記文書を参照ください。

<https://download.samba.org/pub/rsync/rsync.1>
--max-size=SIZE

```
--
例 :
[user@localhost ~]$ hsync --max-size=1000 ...
[user@localhost ~]$ hsync --max-size=1.5mb-1 ...
--
```

1.4.33 min-size

```
=====
対応 OS : Linux / Windows
書式 : --min-size=<size>
-----
```

```
size
既定値 : なし
値の範囲 : 数値文字列、単位付きサイズ指定文字列
=====
```

転送するファイルの最小サイズを指定します。

size には、数値文字列もしくは単位 (B, K, M, G, T, P) 付きのサイズ文字列を指定します。サポートする書式は—max-size と同じです。

--

例 :

```
[user@localhost ~]$ hsync --min-size=1000 ...
```

```
[user@localhost ~]$ hsync --min-size=1.5mb-1 ...
```

--

1.4.34 m, prune-empty-dirs

=====
対応 OS : Linux / Windows

書式 : -m | --prune-empty-dirs
=====

空のディレクトリを転送しないように指示します。

include/exclude/filter による不要なルール検査処理を省くなどのために使用します。

--

例 :

```
[user@localhost ~]$ hsync -m ...
```

--

1.4.35 numeric-ids

=====
対応 OS : Linux / Windows

書式 : --numeric-ids
=====

転送先でのファイルの所有者とグループの保持を転送元ファイルの UID/GID の値を使用して行うように指示します。

--

例 :

```
[user@localhost ~]$ hsync --numeric-ids ...
```

--

1.4.36 usermap

=====
対応 OS : Linux / Windows

書式 : --usermap=<mapping>[,<mapping>...]

mapping

既定値 : なし

値の範囲 : 転送元のユーザ名（もしくは UID 値の範囲）と転送先でのユーザ名（もしくは UID 値）の対応関係を記述した文字列
=====

—owner オプションにより転送元の所有者を転送先のファイルで保持する場合に、指定したユーザマッピングを行ってから保持するように指示します。

mapping には、ユーザマッピングを記述した文字列を指定します。転送元にはユーザ名もしくは UID 値の範囲指定が使用できます。転送先にはユーザ名もしくは UID 値が使用できます。書式については下記文書を参照ください。

<https://download.samba.org/pub/rsync/rsync.1>

--usermap=STRING, --groupmap=STRING

--

例 :

```
[user@localhost ~]$ hsync --usermap=0-99:nobody,wayne:admin,*:normal ...
```

--

1.4.37 groupmap

=====
対応 OS : Linux / Windows

書式 : --groupmap=<mapping>[,<mapping>...]

mapping

既定値 : なし

値の範囲 : 転送元のグループ名 (もしくは GID 値の範囲) と転送先でのグループ名 (もしくは GID 値) の対応関係を記述した文字列

=====
—group オプションにより転送元のグループを転送先のファイルで保持する場合に、指定したグループマッピングを行ってから保持するように指示します。

mapping には、グループマッピングを記述した文字列を指定します。転送元にはグループ名もしくは GID 値の範囲指定が使用できます。転送先にはグループ名もしくは GID 値が使用できます。書式については下記文書を参照ください。

<https://download.samba.org/pub/rsync/rsync.1>

--usermap=STRING, --groupmap=STRING

--

例 :

```
[user@localhost ~]$ hsync --groupmap=usr:1,1:usr ...
```

--

1.4.38 chown

=====
対応 OS : Linux / Windows

書式 : --chown=<user>[:<group>]

user

既定値 : なし

値の範囲 : ユーザ名

group

既定値 : なし

値の範囲 : グループ名

指定したユーザ名とグループ名を転送先のファイルの所有者とグループに適用するように指示します。

このオプションは、`—usermap` 及び `—groupmap` オプションを内部的に使用して実現されます。下記のように実行した場合はどちらも同じ結果となります。

```
[user@localhost ~]$ hsync --chown=foo:bar ...
[user@localhost ~]$ hsync --usermap=*:foo --groupmap=*:bar ...
```

1.4.39 `-I, ignore-times`

対応 OS : Linux / Windows

書式 : `-I` | `--ignore-times`

サイズと更新日時が同じでもファイルをコピーするように指示します。

--

例 :

```
[user@localhost ~]$ hsync -I ...
```

--

1.4.40 `size-only`

対応 OS : Linux / Windows

書式 : `--size-only`

サイズが異なればコピーを行うように指示します。更新日が同じ場合でもコピーされるようになります。

--

例 :

```
[user@localhost ~]$ hsync --size-only ...
```

--

1.4.41 z, compress

=====
対応 OS : Linux / Windows

書式 : -z | --compress
=====

転送するファイルのデータを圧縮するように指示します。

本コマンドでは、ファイルのデータを梱包したメッセージ、ファイルの送信要求などを行う制御メッセージなどサーバと交換するすべてのメッセージを対象に圧縮を行います。

--

例 :

[user@localhost ~]\$ hsync -z ...

--

1.4.42 compress-level

=====
対応 OS : Linux / Windows

書式 : --compress-level=<level>

level

既定値 : 5

値の範囲 : 0 - 9
=====

圧縮レベルを指定します。

--

例 :

[user@localhost ~]\$ hsync --compress-level=5 ...

--

1.4.43 f, filter

=====
対応 OS : Linux / Windows

書式 : -f <filter-desc> | --filter=<filter-desc>

filter-desc

既定値 : なし

値の範囲 : フィルタ文字列
=====

ファイル選択フィルタを追加します。フィルタは（後述する—exclude, —exclude-from, —include, —include-from 含め）指定した順番で適用されます。書式、適用ルールなどは下記文書を参照ください。

<https://download.samba.org/pub/rsync/rsync.1>

FILTER RULES

INCLUDE/EXCLUDE PATTERN RULES

MERGE-FILE FILTER RULES

LIST-CLEARING FILTER RULE

ANCHORING INCLUDE/EXCLUDE PATTERNS

PER-DIRECTORY RULES AND DELETE

本版では、以下の機能は未実装です。

- マージフィルタ (merge, dir-merge) （対応予定です）
- exclude/include の更新子(modifier)のうち"/", "C", "s", "r", "p", "x" ("/", "s", "r"は対応予定です)

--

例：

// すべてのディレクトリをたどって.c ファイルをコピー

```
[user@localhost ~]$ hsync --filter="+ */" --filter="+ *.c" --filter="- *"
...
```

// 同じ作用

```
[user@localhost ~]$ hsync --include="*/" --filter="+ *.c" --exclude="*"
...
```

--

1.4.44 exclude

=====

対応 OS : Linux / Windows

書式 : --exclude=<exclude-desc>

exclude-desc

既定値 : なし

値の範囲 : 除外フィルタ文字列

=====

除外フィルタを追加します。書式、適用ルールなどは下記文書を参照ください。

<https://download.samba.org/pub/rsync/rsync.1>

FILTER RULES

INCLUDE/EXCLUDE PATTERN RULES

本オプションは、前述の **exclude** ルール用の更新子(modifier)は記述できません。また、**+**と先頭に記述した場合は、**include** ルールとなります。**!**と記述した場合は、**clear** ルールとなります。

```
--
例 :
[user@localhost ~]$ hsync --exclude="*.c" ...
--
```

1.4.45 exclude-from

```
=====
対応 OS : Linux / Windows
書式 : --exclude-from=<file>
-----
```

file
既定値 : なし
値の範囲 : 除外ルールを読み込むファイルのパス文字列

```
=====
```

除外ルールを読み込むファイルを追加します。ファイルの内容の書式などは下記文書を参照ください。

<https://download.samba.org/pub/rsync/rsync.1>
FILTER RULES
INCLUDE/EXCLUDE PATTERN RULES

前後に他のフィルタルールが指定された場合は、その間にファイルから読み込んだルールが挿入されます。

```
--
例 :
[user@localhost ~]$ cat .exclude-rule
*
[user@localhost ~]$ hsync --include="*/" --include="*.c" --exclude-from=".
exclude-rule" ...
--
```

1.4.46 include

```
=====
対応 OS : Linux / Windows
書式 : --include=<include-desc>
-----
```

include-desc
既定値 : なし
値の範囲 : 含有フィルタ文字列

含有フィルタを追加します。書式、適用ルールなどは下記文書を参照ください。

<https://download.samba.org/pub/rsync/rsync.1>

FILTER RULES

INCLUDE/EXCLUDE PATTERN RULES

本オプションは、前述の **include** ルール用の更新子(**modifier**)は記述できません。
また、“-”と先頭に記述した場合は、**exclude** ルールとなります。“!”と記述した場合は、**clear** ルールとなります。

--

例 :

```
[user@localhost ~]$ hsync --include="*.c" ...
```

--

1.4.47 include-from

=====
対応 OS : Linux / Windows

書式 : --include-from=<file>

file

既定値 : なし

値の範囲 : 含有ルールを読み込むファイルのパス文字列

=====
含有ルールを読み込むファイルを追加します。ファイルの内容の書式などは下記文書を参照ください。

<https://download.samba.org/pub/rsync/rsync.1>

FILTER RULES

INCLUDE/EXCLUDE PATTERN RULES

前後に他のフィルタルールが指定された場合は、その間にファイルから読み込んだルールが挿入されます。

--

例 :

```
[user@localhost ~]$ cat .include-rule
```

```
*/
```

```
*.c
```

```
[user@localhost ~]$ hsync --include-from=".include-rule" --exclude="*" ...
```

--

1.4.48 log-file

=====
対応 OS : Linux / Windows

書式 : `--log-file=<file>`

file

既定値 : なし

値の範囲 : ログを出力ファイルのパス文字列

=====

コピーの進行状況などのログをファイルに出力するように指示します。ログは指定したファイルへ追加的に記録されます（標準出力にもログが出力されます）。

--

例 :

```
[user@localhost ~]$ hsync --log-file=hsync.log --exclude="hsync.log" ...
```

--

1.4.49 bwlimit

=====

対応 OS : Linux / Windows

書式 : `--bwlimit=<bw-desc>`

bw-desc

既定値 : なし

値の範囲 : 数値文字列（バイト単位）、単位付きスループット指定文字列（バイト単位）

=====

送受信の帯域制限を指定します。

bw-desc には、数値文字列もしくは単位（B, K, M, G, T, P）付きのスループット文字列を指定します。値はバイト単位です。サポートする書式は—**max-size** と同じです。

本オプションの帯域制限は、設定項目の **MaxSendRate/MaxReceiveRate** と同じ方式で行われます。

--

例 :

```
[user@localhost ~]$ hsync --bwlimit=1000000 ...
```

```
[user@localhost ~]$ hsync --bwlimit=1mb ...
```

--

1.4.50 stop-after

=====

対応 OS : Linux / Windows

書式 : `--stop-after=<mins-desc>`

mins-desc

既定値 : -1

値の範囲 : -1, 0 - 符号付き整数の最大値 (分単位)

=====

実行時間の上限を指定します。

mins-desc には、実行時間の上限を分単位で数値で指定します。-1 を指定した場合は上限を設定しません。

指定した時間経過してもコピーが完了しない場合は、コピーを中断します。

--

例 :

```
[user@localhost ~]$ hsync --stop-after=30 ...
```

--

1.4.51 -V, version

=====

対応 OS : Linux / Windows

書式 : -V | --version

=====

hsync コマンドのバージョンを表示します。

--

例 :

```
[user@localhost ~]$ hsync -V
hsync client (hsync) 1.3.0_42 / Linux (HpFP2 2.0.0.91_16)
```

--

1.4.52 config-test

=====

対応 OS : Linux / Windows

書式 : --config-test

=====

hsync コマンドの入力パラメータ及び設定情報を出力します。

--

例 :

```
[user@localhost ~]$ hsync --config-test
...
```

Number of Logical Processors : 2

Number of Physical Processors : 2

Command parameters

```
info          : -
debug         : -
checksum      : disable
archive       : disable
recursive     : disable
relative      : disable
no-implied-dirs : disable
backup        : disable
backup-dir    : -
suffix        : -
update        : disable
inplace       : disable
dirs          : disable
mkpath        : disable
links         : disable
copy-links    : disable
copy-unsafe-links : disable
safe-links    : disable
copy-dirlinks : disable
keep-dirlinks : disable
perms         : disable
executability : disable
chmod         : -
owner         : disable
group         : disable
devices       : disable
specials      : disable
-D            : disable
times         : disable
atimes        : disable
open-notime   : disable
crtimes       : disable
omit-dir-times : disable
omit-link-times : disable
super         : disable
sparse        : disable
dry-run       : disable
whole-file    : disable
existing       : disable
ignore-existing : disable
delete        : disable
delete-before : disable
delete-during : disable
delete-delay  : disable
delete-after  : disable
delete-excluded : disable
ignore-errors : disable
```

```
force          : disable
max-delete     : -
max-size       : -
min-size       : -
partial        : disable
partial-dir    : -
delay-updates  : disable
prune-empty-dirs : disable
numeric-ids    : disable
usermap        : -
groupmap       : -
chown          : -
ignore-times   : disable
size-only      : disable
modify-window  : 0
temp-dir       : -
compress       : disable
compress-level : 5
filter         : -
exclude        : -
exclude-from   : -
include        : -
include-from   : -
files-from     : -
from0          : disable
protect-args   : disable
copy-as        : -
stats          : disable
8-bit-output   : disable
human-readable : disable
progress       : disable
-P            : disable
itemize-changes : disable
out-format     : -
log-file       : -
bwlimit        : -
stop-at        : -
stop-after     : -1
write-batch    : -
only-write-batch : -
read-batch     : -
udp            : disable (use TCP)
version        : disable
config-test    : enable
help           : disable
user           : disable
password       : disable
```

HCP tools コマンドリファレンス

```
config-file      : disable
log-file         : disable
stat-log-file    : disable
multi-run        : disable
```

Configuration parameters

```
PubkeyAuthentication      : yes
WinLogonUserAuthentication : yes
PAMAuthentication        : yes
LocalPasswordAuthentication : yes
RequireServerCertificateSecurity : yes
RejectFallbackServerCertificateSecurity : yes
IgnoreCertificateCNInvalid : yes
IgnoreCertificateDateInvalid : no
IgnoreUnknownCA           : no
IgnoreRevocation          : no
CompressLevel             : -1
HeaderCompress            : yes
ContentCompress           : yes
OCSPRevocationEnabled     : yes
StrictHostKeyChecking     : ask
ApplicationStatLog        : no Rotation[size=no:0:0, pattern=no:]
TransportStatLog          : no Rotation[size=no:0:0, pattern=no:]
PrivateKeySearchDir       : - [/etc/hcp/keys]
PrivateKeyFile            : - [~/.hcp/id_rsa]
CACertificateFile         : - [/etc/hcp/cacert.pem]
CACertificatePath         : -
CARevocationFile         : - [/etc/hcp/crl.pem]
CARevocationPath         : -
ProtocolVersion           : 2
MaxConcurrentThread       : 0
UDPTransportExtensionBufferSize : 2147483648
MaxRequestFileEntryAtOnce : 50
MaxReceiveFileSize       : 107374182400
MaxSendFileSize          : 107374182400
MaxBufferSize            : 1073741824
MaxReceiveRate           : 10000000000
MaxSendRate              : 10000000000
InitHeaderBlockSize      : 51200
InitContentBlockSize     : 51200
MaxHeaderBlockSize       : 51200
MaxContentBlockSize      : 51200
TransportTimeout         : 180
FileLock                 : yes
FileLockTrials           : 0
```

```

FileLockTrialInterval      : 3
AcceptableCryptMethod      : AES256/CBC AES128/CBC [Intel:AES-NI=yes]
AcceptableDigestMethod     : SHA256 SHA160
DisableDataIntegrityChecking : no
AcceptDataIntegrityCheckingOnRejection : yes
TransportCharEncoding      : UTF8
HostEncoding               : UTF8
ApplicationLog              : INFO no Rotation[size=no:0:0, pattern=no:]
PubkeyAuthenticationPrior  : yes
ApplicationStatLogSecurityEx : yes

```

1.4.53 h, help

```

=====
対応 OS : Linux / Windows
書式 : -h | --help
=====

```

hsync コマンドのヘルプを表示します。

このオプションは、短縮名で指定した場合は、転送元と転送先のパスを指定しない場合に動作します。

```

--
例 :
[user@localhost ~]$ hsync -h
--

```

1.4.54 hcp-log-file

hcp コマンドの—log-file オプション参照

1.4.55 hcp-stat-log-file

hcp コマンドの—stat-log-file オプション参照

1.5 hrm

```

Usage: hrm [OPTION]... [USER@]HOST:[PORT:]TARGET
       or: hrm [OPTION]... --host=HOST [--port=PORT] [--user=USER] TARGET

```

hrm コマンドは、リモート（サーバ、hcpd サーバ）上のファイルを削除するコマンドです。

短縮名	オプション名	概略

f	force	確認プロンプトを表示せずに削除
R	recursive	再帰的にファイルを削除
d	dir	空ディレクトリを削除
i		削除毎に確認プロンプトを表示する
I		一度だけ確認プロンプトを表示する
n	no-diskio	ディスク I/O 操作を行わず通信プロトコルのネットワーク I/O 性能を計測する
	hpfp	HpFP プロトコルを使用する
	wss	WebSocket プロトコル(SSL/TLS)を使用する
V	version	バージョンを表示する
	config-test	入力パラメータ及び設定ファイルのロード結果を表示する
h	help	ヘルプを表示する
	user	ユーザ名
	password	パスワード
	host	リモートホスト指定
	port	リモートポート指定
	config-file	設定ファイル
	log-file	ログファイル
	stat-log-file	統計ログファイル
	udp	UDP (HpFP)プロトコルを使用する (ポート分離型)
	ws	WebSocket プロトコル (平文通信) を使用する
	wss-no-check-certificate	WebSocket SSL/TLS サーバ証明書検証無効化
	ws-proxy	WebSocket プロキシサーバ指定
	hpfp-cong	HpFP プロトコルの輻輳制御モードを指定
	hpfp-mss	HpFP プロトコルの MSS 指定
	hpfp-sndbuf	HpFP プロトコルの送信バッファサイズ指定
	hpfp-rcvbuf	HpFP プロトコルの受信バッファサイズ指定
	hcp-out	実行記録を出力するファイルを指定する
	investigation	調査モード

	multi-run	多重起動モード
--	-----------	---------

次のオプションは `hcpd` コマンドの説明をご参照ください。

`--investigation`

次のオプションは `hcp` コマンドの説明をご参照ください。

`--hpfp`, `--wss`, `--user`, `--password`, `--config-file`, `--log-file`, `--stat-log-file`, `--udp`, `--ws`, `--wss-no-check-certificate`, `--ws-proxy`, `--hpfp-cong`, `-hpfp-mss`, `--hpfp-sndbuf`, `--hpfp-rcvbuf`, `--hcp-out`, `--multi-run`

1.5.1 f, force

=====
対応 OS : Linux / Windows

書式 : `-f` | `--force`
=====

確認プロンプトを表示せずに削除を実行します。パーミッション等の理由で確認が必要な場合でも確認せずに削除を実行します。

--

例 :

`[user@localhost ~]$ hrm -f ...`

--

1.5.2 R, recursive

=====
対応 OS : Linux / Windows

書式 : `-R` | `--recursive`
=====

再帰的にファイルを削除します。

--

例 :

`[user@localhost ~]$ hrm -R ...`

--

1.5.3 d, dir

=====
対応 OS : Linux / Windows

書式 : `-d` | `--dir`
=====

空ディレクトリも削除します。

--

例 :

```
[user@localhost ~]$ hrm -d ...
```

--

1.5.4 i

=====

対応 OS : Linux / Windows

書式 : -i

=====

削除毎に確認プロンプトを表示します。

--

例 :

```
[user@localhost ~]$ hrm -i ...
```

--

1.5.5 I

=====

対応 OS : Linux / Windows

書式 : -I

=====

一度確認プロンプトを表示した後は、プロンプトを表示しない様にします。

--

例 :

```
[user@localhost ~]$ hrm -I ...
```

--

1.5.6 V, version

=====

対応 OS : Linux / Windows

書式 : -V | --version

=====

hrm コマンドのバージョンを表示します。

--

例 :

```
[user@localhost ~]$ hrm -V  
hrm client (hrm) 1.3.0_42 / Linux (HpFP2 2.0.0.91_16)
```

--

1.5.7 config-test

=====
対応 OS : Linux / Windows

書式 : --config-test
=====

hrm コマンドの入力パラメータ及び設定情報を出力します。

--

例 :

[user@localhost ~]\$ hrm -t

...

Number of Logical Processors : 2

Number of Physical Processors : 2

Command parameters

force : disable
recursive : disable
dir : disable
prompt_every : disable
prompt_once : disable
no-diskio : disable (0, 0)
host : disable
port : disable
hpfp : disable (TCP used)
ws : disable (TCP used)
ws-proxy : -
wss-no-check-certificate : disable
version : disable
config-test : enable
help : disable
user : disable
password : disable
config-file : disable
log-file : disable
stat-log-file : disable
hcp-out : - [.hcp.out]
multi-run : disable

Target:

Configuration parameters

PubkeyAuthentication : yes
WinLogonUserAuthentication : yes
PAMAuthentication : yes

```

LocalPasswordAuthentication      : yes
RequireServerCertificateSecurity : yes
RejectFallbackServerCertificateSecurity : yes
IgnoreCertificateCNInvalid       : no
IgnoreCertificateDateInvalid     : no
IgnoreUnknownCA                  : no
IgnoreRevocation                 : no
WSSIgnoreCertificateCNInvalid    : no
WSSIgnoreCertificateDateInvalid  : no
WSSIgnoreUnknownCA              : no
WSSIgnoreRevocation              : no
WSSOptions                       : -
WSSCipherSuites                  : -
WSSCipherList                    : -
CompressLevel                    : -1
HeaderCompress                   : yes
ContentCompress                  : yes
OCSPRevocationEnabled            : yes
StrictHostKeyChecking            : ask
ApplicationStatLog                : yes Rotation[size=no:0:0, pattern=no:]
o:]
TransportStatLog                 : no Rotation[size=no:0:0, pattern=no:]
PrivateKeySearchDir              : - [/etc/hcp/keys]
PrivateKeyFile                    : - [~/.hcp/id_rsa]
CACertificateFile                 : - [/etc/hcp/cacert.pem]
CACertificatePath                 : -
CARevocationFile                 : - [/etc/hcp/crl.pem]
CARevocationPath                 : -
WSSCACertificateFile             : -
ProtocolVersion                  : 2
MaxConcurrentThread              : 0
UDPTransportExtensionBufferSize  : 2147483648
TCPTransportSocketSendBuffer     : 0
MaxRequestFileEntryAtOnce        : 50
MaxReceiveFileSize               : unlimited
MaxSendFileSize                  : unlimited
MaxBufferSize                    : 1073741824
MaxReceiveRate                   : 10000000000
MaxSendRate                      : 10000000000
InitHeaderBlockSize              : 51200
InitContentBlockSize             : 1048576
MaxHeaderBlockSize               : 51200
MaxContentBlockSize              : 1048576
TransportTimeout                 : 180
FileLock                         : no
FileLockTrials                   : 0
FileLockTrialInterval            : 3

```

```
AcceptableCryptMethod      : AES256/CTR/VMAC AES256/CBC AES128/CBC
[Intel:AES-NI=yes]
AcceptableDigestMethod     : MM128 SHA256 SHA160
DisableDataIntegrityChecking : no
AcceptDataIntegrityCheckingOnRejection : yes
TransportCharEncoding      : UTF8
HostEncoding               : UTF8
ApplicationLog              : INFO no Rotation[size=no:0:0, pattern
=no:]
PubkeyAuthenticationPrior  : yes
ApplicationStatLogSecurityEx : yes
--
```

1.5.8 h, help

```
=====
対応 OS : Linux / Windows
書式 : -h | --help
=====
```

hrm コマンドのヘルプを表示します。

```
--
例 :
[user@localhost ~]$ hrm -h
--
```

1.5.9 host

```
=====
対応 OS : Linux / Windows
書式 : --host=<remote-host>
-----
```

```
remote-host
既定値 : なし
値の範囲 : IP アドレス もしくは ホスト名
=====
```

接続先のリモートホストを指定します。削除ターゲットのパスに適用されます。
このパラメータを指定した場合、このパスのホスト名を省略することができます。

```
--
例 :
[user@localhost ~]$ hrm --host=192.168.100.100 ...
--
```

1.5.10 port

=====
対応 OS : Linux / Windows

書式 : --port=<remote-port>

remote-port

既定値 : なし

値の範囲 : ポート番号
=====

接続先のリモートホストのサービスポート番号を指定します。削除ターゲットのパスに適用されます。

--

例 :

[user@localhost ~]\$ hrm --port=1874 ...

--

1.5.11 hcp-out

=====
対応 OS : Linux / Windows

書式 : --hcp-out=<output-path>

output-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列
=====

ファイル削除の実行記録を出力するファイルを指定します。

--

例 :

TARGET 127.0.0.1:11111:/home/user/Desktop/hcp_dst5

OK 0000 DE 00000001 ./

OK 0000 FR 00000002 ./file1.txt

OK 0000 DX 00000001 ./

EXIT 0 REASON 0000

--

FR (File Remove)は、ファイルの削除処理を表します。DE(Directory Enter)は、i オプションを指定したときに行われるディレクトリに対する削除を開始するかの確認処理を表します。DX(Directory eXit)は、i オプションに指定したときに行われるディレクトリに対する削除確認処理を表します。

--

例 :

```
[user@localhost ~]$ hrm -v ...
--
```

1.6 hcp-ls

Usage: hcp-ls [OPTION]... [USER@]HOST[:PORT][:FILE] [:FILE]...
 or: hcp-ls [OPTION]... --host=HOST [--port=PORT] [--user=USER] [FILE]...

hcp-ls コマンドは、リモート（サーバ、hcpd サーバ）上のファイル一覧を取得するコマンドです。

短縮名	オプション名	概略
q	query-cmdname	コマンド名問合せフラグ
o	cmd-options	コマンドオプション
	hpfp	HpFP プロトコルを使用する
	wss	WebSocket プロトコル(SSL/TLS)を使用する
V	version	バージョンを表示する
	config-test	入力パラメータ及び設定ファイルのロード結果を表示する
h	help	ヘルプを表示する
	user	ユーザ名
	password	パスワード
	host	リモートホスト指定
	port	リモートポート指定
	config-file	設定ファイル
	log-file	ログファイル
	stat-log-file	統計ログファイル
	udp	UDP (HpFP)プロトコルを使用する（ポート分離型）
	ws	WebSocket プロトコル（平文通信）を使用する
	wss-no-check-certificate	WebSocket SSL/TLS サーバ証明書検証無効化
	ws-proxy	WebSocket プロキシサーバ指定
	hpfp-cong	HpFP プロトコルの輻輳制御モードを指定

	hpf-mss	HpFP プロトコルの MSS 指定
	hpf-sndbuf	HpFP プロトコルの送信バッファサイズ指定
	hpf-rcvbuf	HpFP プロトコルの送信バッファサイズ指定
	hcp-out	実行記録を出力するファイルを指定する
	investigation	調査モード
	multi-run	多重起動モード

次のオプションは hcpd コマンドの説明をご参照ください。

--investigation

次のオプションは hcp コマンドの説明をご参照ください。

--hpf, --wss, --user, --password, --config-file, --log-file, --stat-log-file, --udp, --ws, --wss-no-check-certificate, --ws-proxy, --hpf-cong, --hpf-mss, --hpf-sndbuf, --hpf-rcvbuf, --hcp-out, --multi-run

1.6.1 q, query-cmdname

対応 OS : Linux / Windows

書式 : -q | --query-cmdname

サーバ上で実行されるリストコマンドの名前を問合せます。以下の何れかが表示されます。

- ls (Linux)
- dir (Windows)

--

例 :

```
[user@localhost ~]$ hcp-ls -q ...
```

--

1.6.2 o, cmd-options

対応 OS : Linux / Windows

書式 : -o <options> | --cmd-options=<options>

options

既定値 : なし

値の範囲 : ls もしくは dir コマンドの引数オプション

コマンドのオプションを指定します。空白もしくはハイフン“-“を含む場合は、クオートします。

```
--
例 :
[user@localhost ~]$ hcp-ls -o "-al" ...
--
```

1.6.3 V, version

```
=====
対応 OS : Linux / Windows
書式 : -V | --version
=====
```

hcp-ls コマンドのバージョンを表示します。

```
--
例 :
[user@localhost ~]$ hcp-ls -V
hcp-ls client (hcp-ls) 1.3.0_42 / Linux (HpFP2 2.0.0.91_16)
--
```

1.6.4 config-test

```
=====
対応 OS : Linux / Windows
書式 : --config-test
=====
```

hcp-ls コマンドの入力パラメータ及び設定情報を出力します。

```
--
例 :
[user@localhost ~]$ hcp-ls -t
...

Number of Logical Processors : 2
Number of Physical Processors : 2

Command parameters
query-cmdname      : disable
cmd-options       : -
host              : disable
port              : disable
hpfp              : disable (TCP used)
ws                : disable (TCP used)
ws-proxy          : -
```

HCP tools コマンドリファレンス

```
wss-no-check-certificate : disable
version                  : disable
config-test              : enable
help                     : disable
user                     : disable
password                 : disable
config-file              : disable
log-file                 : disable
stat-log-file            : disable
hcp-out                   : - [.hcp.out]
multi-run                 : disable
```

Configuration parameters

```
PubkeyAuthentication      : yes
WinLogonUserAuthentication : yes
PAMAuthentication        : yes
LocalPasswordAuthentication : yes
RequireServerCertificateSecurity : yes
RejectFallbackServerCertificateSecurity : yes
IgnoreCertificateCNInvalid : no
IgnoreCertificateDateInvalid : no
IgnoreUnknownCA           : no
IgnoreRevocation           : no
WSSIgnoreCertificateCNInvalid : no
WSSIgnoreCertificateDateInvalid : no
WSSIgnoreUnknownCA        : no
WSSIgnoreRevocation        : no
WSSOptions                 : -
WSSCipherSuites            : -
WSSCipherList              : -
CompressLevel              : -1
HeaderCompress             : yes
ContentCompress            : yes
OCSPRevocationEnabled      : yes
StrictHostKeyChecking      : ask
ApplicationStatLog         : yes Rotation[size=no:0:0, pattern=no:]
o:]
TransportStatLog           : no Rotation[size=no:0:0, pattern=no:]
PrivateKeySearchDir        : - [/etc/hcp/keys]
PrivateKeyFile              : - [~/.hcp/id_rsa]
CACertificateFile           : - [/etc/hcp/cacert.pem]
CACertificatePath          : -
CARevocationFile           : - [/etc/hcp/crl.pem]
CARevocationPath           : -
WSSCACertificateFile       : -
```



```

ProtocolVersion           : 2
MaxConcurrentThread       : 0
UDPTransportExtensionBufferSize : 2147483648
TCPTransportSocketSendBuffer : 0
MaxRequestFileEntryAtOnce : 50
MaxReceiveFileSize        : unlimited
MaxSendFileSize           : unlimited
MaxBufferSize             : 1073741824
MaxReceiveRate            : 10000000000
MaxSendRate               : 10000000000
InitHeaderBlockSize       : 51200
InitContentBlockSize      : 1048576
MaxHeaderBlockSize        : 51200
MaxContentBlockSize       : 1048576
TransportTimeout          : 180
FileLock                  : no
FileLockTrials            : 0
FileLockTrialInterval     : 3
AcceptableCryptMethod     : AES256/CTR/VMAC AES256/CBC AES128/CBC
[Intel:AES-NI=yes]
AcceptableDigestMethod    : MM128 SHA256 SHA160
DisableDataIntegrityChecking : no
AcceptDataIntegrityCheckingOnRejection : yes
TransportCharEncoding     : UTF8
HostEncoding              : UTF8
ApplicationLog             : INFO no Rotation[size=no:0:0, pattern
=no:]
PubkeyAuthenticationPrior : yes
ApplicationStatLogSecurityEx : yes
--

```

1.6.5 h, help

```

=====
対応 OS : Linux / Windows
書式 : -h | --help
=====

```

hcp-ls コマンドのヘルプを表示します。

```

--
例 :
[user@localhost ~]$ hcp-ls -h
--

```

1.6.6 host

```

=====
対応 OS : Linux / Windows

```

書式 : `--host=<remote-host>`

remote-host

既定値 : なし

値の範囲 : IP アドレス もしくは ホスト名

=====

接続先のリモートホストを指定します。リスト対象を指定するパスに適用されます。このパラメータを指定した場合、このパスのホスト名を省略することができます。

--

例 :

`[user@localhost ~]$ hcp-ls --host=192.168.100.100 ...`

--

1.6.7 port

=====

対応 OS : Linux / Windows

書式 : `--port=<remote-port>`

remote-port

既定値 : なし

値の範囲 : ポート番号

=====

接続先のリモートホストのサービスポート番号を指定します。リスト対象を指定するパスに適用されます。

--

例 :

`[user@localhost ~]$ hcp-ls --port=1874 ...`

--

1.6.8 hcp-out

=====

対応 OS : Linux / Windows

書式 : `--hcp-out=<output-path>`

output-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

=====

ファイルリスト取得の実行記録を出力するファイルを指定します。リストの実行結果は常に標準出力へ出力されます。

```
--
例：
FILE 127.0.0.1:11111:/home/user/Desktop/hcp_dst5
--

--
例：
[user@localhost ~]$ hcp-ls -v ...
--
```

1.7 hmkdir

Usage: hmkdir [OPTION]... [USER@]HOST[:PORT]:DIRECTORY [:DIRECTORY]...
 or: hmkdir [OPTION]... --host=HOST [--port=PORT] [--user=USER] DIRECTORY...

hmkdir コマンドは、リモート（サーバ、hcpd サーバ）上でディレクトリを作成するコマンドです。

短縮名	オプション名	概略
p	parents	中間ディレクトリ作成フラグ
	hfp	HpFP プロトコルを使用する
	wss	WebSocket プロトコル(SSL/TLS)を使用する
V	version	バージョンを表示する
	config-test	入力パラメータ及び設定ファイルのロード結果を表示する
h	help	ヘルプを表示する
	user	ユーザ名
	password	パスワード
	host	リモートホスト指定
	port	リモートポート指定
	config-file	設定ファイル
	log-file	ログファイル
	stat-log-file	統計ログファイル
	udp	UDP (HpFP)プロトコルを使用する（ポート分離

		型)
	ws	WebSocket プロトコル（平文通信）を使用する
	wss-no-check-certificate	WebSocket SSL/TLS サーバ証明書検証無効化
	ws-proxy	WebSocket プロキシサーバ指定
	hpfp-cong	HpFP プロトコルの輻輳制御モードを指定
	hpfp-mss	HpFP プロトコルの MSS 指定
	hpfp-sndbuf	HpFP プロトコルの送信バッファサイズ指定
	hpfp-rcvbuf	HpFP プロトコルの送信バッファサイズ指定
	hcp-out	実行記録を出力するファイルを指定する
	investigation	調査モード
	multi-run	多重起動モード

次のオプションは hcpd コマンドの説明をご参照ください。

--investigation

次のオプションは hcp コマンドの説明をご参照ください。

--hpfp, --wss, --user, --password, --config-file, --log-file, --stat-log-file, --udp, --ws, --wss-no-check-certificate, --ws-proxy, --hpfp-cong, --hpfp-mss, --hpfp-sndbuf, --hpfp-rcvbuf, --hcp-out, --multi-run

1.7.1 p, parents

指定されたパスの中間ディレクトリが存在しない場合に、それらのディレクトリも作成するように指示します。

--

例：

```
[user@localhost ~]$ hmkdir -p ...
```

--

1.7.2 V, version

```
=====
対応 OS : Linux / Windows
書式 : -V | --version
=====
```

hmkdir コマンドのバージョンを表示します。

--

例 :

```
[user@localhost ~]$ hmkdir -h
hmkdir client (hmkdir) 1.3.0_42 / Linux (HpFP2 2.0.0.91_16)
```

--

1.7.3 config-test

=====
対応 OS : Linux / Windows

書式 : --config-test
=====

hmkdir コマンドの入力パラメータ及び設定情報を出力します。

--

例 :

```
[user@localhost ~]$ hmkdir -t
...
```

Number of Logical Processors : 2

Number of Physical Processors : 2

Command parameters

```
parents      : disable
host         : disable
port         : disable
hpfp         : disable (TCP used)
ws           : disable (TCP used)
ws-proxy     : -
wss-no-check-certificate : disable
version      : disable
config-test  : enable
help         : disable
user         : disable
password     : disable
config-file  : disable
log-file     : disable
stat-log-file : disable
hcp-out      : - [.hcp.out]
multi-run    : disable
```

Configuration parameters

```
PubkeyAuthentication      : yes
WinLogonUserAuthentication : yes
PAMAuthentication        : yes
```

```

LocalPasswordAuthentication      : yes
RequireServerCertificateSecurity : yes
RejectFallbackServerCertificateSecurity : yes
IgnoreCertificateCNInvalid       : no
IgnoreCertificateDateInvalid     : no
IgnoreUnknownCA                 : no
IgnoreRevocation                : no
WSSIgnoreCertificateCNInvalid    : no
WSSIgnoreCertificateDateInvalid  : no
WSSIgnoreUnknownCA              : no
WSSIgnoreRevocation              : no
WSSOptions                      : -
WSSCipherSuites                 : -
WSSCipherList                   : -
CompressLevel                   : -1
HeaderCompress                  : yes
ContentCompress                 : yes
OCSPRevocationEnabled           : yes
StrictHostKeyChecking            : ask
ApplicationStatLog               : yes Rotation[size=no:0:0, pattern=no:]
o:]
TransportStatLog                : no Rotation[size=no:0:0, pattern=no:]
PrivateKeySearchDir              : - [/etc/hcp/keys]
PrivateKeyFile                   : - [~/.hcp/id_rsa]
CACertificateFile                : - [/etc/hcp/cacert.pem]
CACertificatePath                : -
CARevocationFile                : - [/etc/hcp/crl.pem]
CARevocationPath                : -
WSSCACertificateFile            : -
ProtocolVersion                 : 2
MaxConcurrentThread              : 0
UDPTransportExtensionBufferSize : 2147483648
TCPTransportSocketSendBuffer    : 0
MaxRequestFileEntryAtOnce       : 50
MaxReceiveFileSize              : unlimited
MaxSendFileSize                 : unlimited
MaxBufferSize                   : 1073741824
MaxReceiveRate                  : 10000000000
MaxSendRate                     : 10000000000
InitHeaderBlockSize             : 51200
InitContentBlockSize            : 1048576
MaxHeaderBlockSize              : 51200
MaxContentBlockSize             : 1048576
TransportTimeout                : 180
FileLock                        : no
FileLockTrials                  : 0
FileLockTrialInterval           : 3

```

```

AcceptableCryptMethod      : AES256/CTR/VMAC AES256/CBC AES128/CBC
[Intel:AES-NI=yes]
AcceptableDigestMethod     : MM128 SHA256 SHA160
DisableDataIntegrityChecking : no
AcceptDataIntegrityCheckingOnRejection : yes
TransportCharEncoding      : UTF8
HostEncoding               : UTF8
ApplicationLog              : INFO no Rotation[size=no:0:0, pattern
=no:]
PubkeyAuthenticationPrior  : yes
ApplicationStatLogSecurityEx : yes
--

```

1.7.4 h, help

```

=====
対応 OS : Linux / Windows
書式 : -h | --help
=====

```

hmkdir コマンドのヘルプを表示します。

```

--
例 :
[user@localhost ~]$ hmkdir -h
--

```

1.7.5 host

```

=====
対応 OS : Linux / Windows
書式 : --host=<remote-host>
-----
remote-host
既定値 : なし
値の範囲 : IP アドレス もしくは ホスト名
=====

```

接続先のリモートホストを指定します。ディレクトリを指定するパスに適用されます。このパラメータを指定した場合、このパス中のホスト名を省略することができます。

```

--
例 :
[user@localhost ~]$ hmkdir --host=192.168.100.100
--

```

1.7.6 port

=====
対応 OS : Linux / Windows

書式 : --port=<remote-port>

remote-port

既定値 : なし

値の範囲 : ポート番号
=====

接続先のリモートホストのサービスポート番号を指定します。リディレクトリを指定するパスに適用されます。

--

例 :

```
[user@localhost ~]$ hmkdir --port=1874
```

--

1.7.7 hcp-out

=====
対応 OS : Linux / Windows

書式 : --hcp-out=<output-path>

output-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列
=====

ディレクトリ作成の実行記録を出力するファイルを指定します。

--

例 :

```
[user@localhost ~]$ hmkdir -v ...  
DIR0 127.0.0.1:1874:/home/user/Desktop/hcp_mkdir14  
DIR1 127.0.0.1:1874:/home/user/Desktop/hcp_mkdir15  
OK 0000 DC 00000001 /home/user/Desktop/hcp_mkdir14  
OK 0000 DC 00000002 /home/user/Desktop/hcp_mkdir15  
EXIT 0 REASON 0000  
[user@localhost ~]$
```

--

--

例 :

```
[user@localhost ~]$ hmkdir -v ...
```

--

1.8 hpwd

Usage: hpwd [OPTION]... [USER@]HOST[:PORT]

hpwd コマンドは、リモート（サーバ、hcpd サーバ）上の作業ディレクトリを表示するコマンドです。

短縮名	オプション名	概略
L	logical	論理パス取得
P	physical	物理パス取得（シンボリックリンク解決）
	hpfp	HpFP プロトコルを使用する
	wss	WebSocket プロトコル(SSL/TLS)を使用する
V	version	バージョンを表示する
	config-test	入力パラメータ及び設定ファイルのロード結果を表示する
h	help	ヘルプを表示する
	user	ユーザ名
	password	パスワード
	config-file	設定ファイル
	log-file	ログファイル
	stat-log-file	統計ログファイル
	udp	UDP (HpFP)プロトコルを使用する（ポート分離型）
	ws	WebSocket プロトコル（平文通信）を使用する
	wss-no-check-certificate	WebSocket SSL/TLS サーバ証明書検証無効化
	ws-proxy	WebSocket プロキシサーバ指定
	hpfp-cong	HpFP プロトコルの輻輳制御モードを指定
	hpfp-mss	HpFP プロトコルの MSS 指定
	hpfp-sndbuf	HpFP プロトコルの送信バッファサイズ指定
	hpfp-rcvbuf	HpFP プロトコルの受信バッファサイズ指定
	investigation	調査モード
	multi-run	多重起動モード

次のオプションは `hcpd` コマンドの説明をご参照ください。

`--investigation`

次のオプションは `hcp` コマンドの説明をご参照ください。

`--hpfp`, `--wss`, `--user`, `--password`, `--config-file`, `--log-file`, `--stat-log-file`, `--udp`, `--ws`, `--wss-no-check-certificate`, `--ws-proxy`, `--hpfp-cong`, `--hpfp-mss`, `--hpfp-sndbuf`, `--hpfp-rcvbuf`, `--hcp-out`, `--multi-run`

1.8.1 L, logical

=====
対応 OS : Linux / Windows

書式 : `-L` | `--logical`
=====

論理パスを返す様に指示します。`logical` オプション及び `physical` オプション共に指定されない場合は、本オプションが指定されたものとして動作します。

--

例 :

`[user@localhost ~]$ hpwd -L ...`

--

1.8.2 P, physical

=====
対応 OS : Linux / Windows

書式 : `-P` | `--physical`
=====

物理パスを返す様に指示します。中間パスに存在するシンボリックリンクは解決されます。

--

例 :

`[user@localhost ~]$ hpwd -P ...`

--

1.8.3 V, version

=====
対応 OS : Linux / Windows

書式 : `-V` | `--version`
=====

`hpwd` コマンドのバージョンを表示します。

--

例 :

```
[user@localhost ~]$ hpwd -V  
hpwd client (hpwd) 1.3.0_42 / Linux (HpFP2 2.0.0.91_16)
```

--

1.8.4 config-test

=====
対応 OS : Linux / Windows

書式 : --config-test
=====

hpwd コマンドの入力パラメータ及び設定情報を出力します。

--

例 :

```
[user@localhost ~]$ hpwd -t  
...
```

Number of Logical Processors : 2

Number of Physical Processors : 2

Command parameters

```
logical      : disable  
physical     : disable  
hpfp         : disable (TCP used)  
ws           : disable (TCP used)  
ws-proxy     : -  
wss-no-check-certificate : disable  
version      : disable  
config-test  : enable  
help         : disable  
user         : disable  
password     : disable  
config-file  : disable  
log-file     : disable  
stat-log-file : disable  
hcp-out      : - [.hcp.out]  
multi-run    : disable
```

Configuration parameters

```
PubkeyAuthentication      : yes  
WinLogonUserAuthentication : yes  
PAMAuthentication        : yes  
LocalPasswordAuthentication : yes
```

```

RequireServerCertificateSecurity : yes
RejectFallbackServerCertificateSecurity : yes
IgnoreCertificateCNInvalid       : no
IgnoreCertificateDateInvalid     : no
IgnoreUnknownCA                 : no
IgnoreRevocation                 : no
WSSIgnoreCertificateCNInvalid    : no
WSSIgnoreCertificateDateInvalid  : no
WSSIgnoreUnknownCA              : no
WSSIgnoreRevocation              : no
WSSOptions                      : -
WSSCipherSuites                 : -
WSSCipherList                   : -
CompressLevel                   : -1
HeaderCompress                  : yes
ContentCompress                  : yes
OCSPRevocationEnabled           : yes
StrictHostKeyChecking            : ask
ApplicationStatLog               : yes Rotation[size=no:0:0, pattern=no:]
o:]
TransportStatLog                 : no Rotation[size=no:0:0, pattern=no:]
PrivateKeySearchDir              : - [/etc/hcp/keys]
PrivateKeyFile                   : - [~/.hcp/id_rsa]
CACertificateFile                : - [/etc/hcp/cacert.pem]
CACertificatePath                : -
CARevocationFile                : - [/etc/hcp/crl.pem]
CARevocationPath                : -
WSSCACertificateFile            : -
ProtocolVersion                  : 2
MaxConcurrentThread              : 0
UDPTransportExtensionBufferSize : 2147483648
TCPTransportSocketSendBuffer    : 0
MaxRequestFileEntryAtOnce       : 50
MaxReceiveFileSize              : unlimited
MaxSendFileSize                 : unlimited
MaxBufferSize                   : 1073741824
MaxReceiveRate                  : 10000000000
MaxSendRate                     : 10000000000
InitHeaderBlockSize             : 51200
InitContentBlockSize            : 1048576
MaxHeaderBlockSize              : 51200
MaxContentBlockSize             : 1048576
TransportTimeout                : 180
FileLock                        : no
FileLockTrials                  : 0
FileLockTrialInterval           : 3
AcceptableCryptMethod           : AES256/CTR/VMAC AES256/CBC AES128/CBC

```

```

[Intel:AES-NI=yes]
AcceptableDigestMethod      : MM128 SHA256 SHA160
DisableDataIntegrityChecking : no
AcceptDataIntegrityCheckingOnRejection : yes
TransportCharEncoding       : UTF8
HostEncoding                 : UTF8
ApplicationLog               : INFO no Rotation[size=no:0:0, pattern
=no:]
PubkeyAuthenticationPrior   : yes
ApplicationStatLogSecurityEx : yes
--

```

1.8.5 h, help

```

=====
対応 OS : Linux / Windows
書式 : -h | --help
=====

```

hpwd コマンドのヘルプを表示します。

```

--
例 :
[user@localhost ~]$ hpwd -h
--

```

1.9 hmv

```

Usage: hmv [OPTION]... [USER@]HOST[:PORT]:SOURCE :DEST
or: hmv [OPTION]... [USER@]HOST[:PORT]:SOURCE [:SOURCE]... :DIRECTORY
or: hmv [OPTION]... --host=HOST [--port=PORT] [--user=USER] SOURCE DEST
T
or: hmv [OPTION]... --host=HOST [--port=PORT] [--user=USER] SOURCE...
DIRECTORY

```

hmv コマンドは、リモート（サーバ、hcpd サーバ）上でファイルもしくはディレクトリを移動するコマンドです。

短縮名	オプション名	概略
f	force	移動先上書き
i	interactive	移動先上書き確認
N	no-overwrite	移動先上書き禁止
	hpfp	HpFP プロトコルを使用する
	wss	WebSocket プロトコル(SSL/TLS)を使用する

V	version	バージョンを表示する
	config-test	入力パラメータ及び設定ファイルのロード結果を表示する
h	help	ヘルプを表示する
	user	ユーザ名
	password	パスワード
	host	リモートホスト指定
	port	リモートポート指定
	config-file	設定ファイル
	log-file	ログファイル
	stat-log-file	統計ログファイル
	udp	UDP (HpFP) プロトコルを使用する (ポート分離型)
	ws	WebSocket プロトコル (平文通信) を使用する
	wss-no-check-certificate	WebSocket SSL/TLS サーバ証明書検証無効化
	ws-proxy	WebSocket プロキシサーバ指定
	hfp-cong	HpFP プロトコルの輻輳制御モードを指定
	hfp-mss	HpFP プロトコルの MSS 指定
	hfp-sndbuf	HpFP プロトコルの送信バッファサイズ指定
	hfp-rcvbuf	HpFP プロトコルの受信バッファサイズ指定
	hcp-out	実行記録を出力するファイルを指定する
	investigation	調査モード
	multi-run	多重起動モード

次のオプションは hcpd コマンドの説明をご参照ください。

--investigation

次のオプションは hcp コマンドの説明をご参照ください。

--hfp, --wss, --user, --password, --config-file, --log-file, --stat-log-file, --udp, --ws, --wss-no-check-certificate, --ws-proxy, --hfp-cong, --hfp-mss, --hfp-sndbuf, --hfp-rcvbuf, --hcp-out, --multi-run

1.9.1 f, force

```
=====
対応 OS : Linux / Windows
書式 : -f | --force
=====
```

移動先にファイルもしくはディレクトリが存在する場合に上書きします。

```
--
例 :
[user@localhost ~]$ hmv -f ...
--
```

1.9.2 i, interactive

```
=====
対応 OS : Linux / Windows
書式 : -i | --interactive
=====
```

移動先にファイルもしくはディレクトリが存在する場合に上書きするか確認します。

```
--
例 :
[user@localhost ~]$ hmv -i ...
--
```

1.9.3 N, no-overwrite

```
=====
対応 OS : Linux / Windows
書式 : -N | --no-overwrite
=====
```

移動先にファイルもしくはディレクトリが存在する場合に上書きせずにスキップします。

```
--
例 :
[user@localhost ~]$ hmv -N ...
--
```

1.9.4 V, version

```
=====
対応 OS : Linux / Windows
書式 : -V | --version
=====
```

hmv コマンドのバージョンを表示します。

--

例 :

```
[user@localhost ~]$ hmv -V
hmv client (hmv) 1.3.0_42 / Linux (HpFP2 2.0.0.91_16)
```

--

1.9.5 config-test

=====
対応 OS : Linux / Windows

書式 : --config-test
=====

hmv コマンドの入力パラメータ及び設定情報を出力します。

--

例 :

```
[user@localhost ~]$ hmv -t
...
```

Number of Logical Processors : 2

Number of Physical Processors : 2

Command parameters

```
force      : disable
interactive : disable
no-overwrite : disable
host       : disable
port       : disable
hpfp       : disable (TCP used)
ws         : disable (TCP used)
ws-proxy   : -
wss-no-check-certificate : disable
version    : disable
config-test : enable
help       : disable
user       : disable
password   : disable
config-file : disable
log-file   : disable
stat-log-file : disable
hcp-out    : - [.hcp.out]
multi-run  : disable
```

Target source:

Target destination:

Configuration parameters

```

PubkeyAuthentication          : yes
WinLogonUserAuthentication    : yes
PAMAuthentication            : yes
LocalPasswordAuthentication   : yes
RequireServerCertificateSecurity : yes
RejectFallbackServerCertificateSecurity : yes
IgnoreCertificateCNInvalid     : no
IgnoreCertificateDateInvalid   : no
IgnoreUnknownCA               : no
IgnoreRevocation              : no
WSSIgnoreCertificateCNInvalid  : no
WSSIgnoreCertificateDateInvalid : no
WSSIgnoreUnknownCA            : no
WSSIgnoreRevocation           : no
WSSOptions                    : -
WSSCipherSuites               : -
WSSCipherList                 : -
CompressLevel                 : -1
HeaderCompress                : yes
ContentCompress               : yes
OCSPRevocationEnabled         : yes
StrictHostKeyChecking         : ask
ApplicationStatLog             : yes Rotation[size=no:0:0, pattern=no:]
TransportStatLog              : no Rotation[size=no:0:0, pattern=no:]
PrivateKeySearchDir           : - [/etc/hcp/keys]
PrivateKeyFile                 : - [~/.hcp/id_rsa]
CACertificateFile              : - [/etc/hcp/cacert.pem]
CACertificatePath              : -
CARevocationFile              : - [/etc/hcp/crl.pem]
CARevocationPath              : -
WSSCACertificateFile          : -
ProtocolVersion               : 2
MaxConcurrentThread           : 0
UDPTransportExtensionBufferSize : 2147483648
TCPTransportSocketSendBuffer  : 0
MaxRequestFileEntryAtOnce     : 50
MaxReceiveFileSize            : unlimited
MaxSendFileSize               : unlimited
MaxBufferSize                 : 1073741824
MaxReceiveRate                : 10000000000
MaxSendRate                   : 10000000000

```

```
InitHeaderBlockSize      : 51200
InitContentBlockSize     : 1048576
MaxHeaderBlockSize       : 51200
MaxContentBlockSize      : 1048576
TransportTimeout         : 180
FileLock                  : no
FileLockTrials            : 0
FileLockTrialInterval    : 3
AcceptableCryptMethod    : AES256/CTR/VMAC AES256/CBC AES128/CBC
[Intel:AES-NI=yes]
AcceptableDigestMethod   : MM128 SHA256 SHA160
DisableDataIntegrityChecking : no
AcceptDataIntegrityCheckingOnRejection : yes
TransportCharEncoding    : UTF8
HostEncoding              : UTF8
ApplicationLog            : INFO no Rotation[size=no:0:0, pattern
=no:]
PubkeyAuthenticationPrior : yes
ApplicationStatLogSecurityEx : yes
--
```

1.9.6 h, help

```
=====
対応 OS : Linux / Windows
書式 : -h | --help
=====
```

hmv コマンドのヘルプを表示します。

```
--
例 :
[user@localhost ~]$ hmv -h
--
```

1.9.7 host

```
=====
対応 OS : Linux / Windows
書式 : --host=<remote-host>
-----
remote-host
既定値 : なし
値の範囲 : IP アドレス もしくは ホスト名
=====
```

接続先のリモートホストを指定します。移動元と移動先を指定するパスに適用されます。このパラメータを指定した場合、このパスのホスト名を省略することができます。

--

例：

```
[user@localhost ~]$ hmv --host=192.168.100.100 ...
```

--

1.9.8 port

=====
対応 OS : Linux / Windows

書式 : --port=<remote-port>

remote-port

既定値 : なし

値の範囲 : ポート番号
=====

接続先のリモートホストのサービスポート番号を指定します。移動元と移動先を指定するパスに適用されます。

--

例：

```
[user@localhost ~]$ hmv --port=1874 ...
```

--

1.9.9 hcp-out

=====
対応 OS : Linux / Windows

書式 : --hcp-out=<output-path>

output-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列
=====

ファイルもしくはディレクトリの移動の実行記録を出力するファイルを指定します。

--

例：

```
[user@localhost ~]$ hmv -v ...
```

```
SRC0 127.0.0.1:1874:/home/user/Desktop/hcp_hmv01.txt
```

```
SRC1 127.0.0.1:1874:/home/user/Desktop/hcp_hmv02.txt
```

```
DST 127.0.0.1:1874:/home/user/Desktop/hcp_hmv_dir
```

```
OK 0000 FM 00000001 /home/user/Desktop/hcp_hmv01.txt
OK 0000 FM 00000002 /home/user/Desktop/hcp_hmv02.txt
EXIT 0 REASON 0000
[user@localhost ~]$
--
```

```
--
例 :
[user@localhost ~]$ hmv -v ...
--
```

1.10 hln

```
Usage: hln [OPTION]... [USER@]HOST[:PORT]:TARGET :LINK_NAME
       or: hln [OPTION]... [USER@]HOST[:PORT]:TARGET [:TARGET]... :DIRECTORY
       or: hln [OPTION]... --host=HOST [--port=PORT] [--user=USER] TARGET LINK_NAME
       or: hln [OPTION]... --host=HOST [--port=PORT] [--user=USER] TARGET...
       DIRECTORY
```

hln コマンドは、リモート（サーバ、**hcpd** サーバ）上でファイルもしくはディレクトリのリンクを作成するコマンドです。本コマンドは、**Windows** サービスではサポートしません。

短縮名	オプション名	概略
f	force	リンク先上書き
i	interactive	リンク先上書き確認
N	no-dereference	シンボリックリンク先ディレクトリの解決を無効にする
s	symbolic	シンボリックリンク作成
L	logical	ターゲットパスを論理名として扱う（シンボリックリンクを解決する）
P	physical	ターゲットパスを物理名として扱う（シンボリックリンクを解決しない）
	hpfp	HpFP プロトコルを使用する
	wss	WebSocket プロトコル(SSL/TLS)を使用する
V	version	バージョンを表示する
	config-test	入力パラメータ及び設定ファイルのロード結果を表示する

h	help	ヘルプを表示する
	user	ユーザ名
	password	パスワード
	host	リモートホスト指定
	port	リモートポート指定
	config-file	設定ファイル
	log-file	ログファイル
	stat-log-file	統計ログファイル
	udp	UDP (HpFP) プロトコルを使用する (ポート分離型)
	ws	WebSocket プロトコル (平文通信) を使用する
	wss-no-check-certificate	WebSocket SSL/TLS サーバ証明書検証無効化
	ws-proxy	WebSocket プロキシサーバ指定
	hpf-p-cong	HpFP プロトコルの輻輳制御モードを指定
	hpf-p-mss	HpFP プロトコルの MSS 指定
	hpf-p-sndbuf	HpFP プロトコルの送信バッファサイズ指定
	hpf-p-rcvbuf	HpFP プロトコルの受信バッファサイズ指定
	hcp-out	実行記録を出力するファイルを指定する
	investigation	調査モード
	multi-run	多重起動モード

次のオプションは hcpd コマンドの説明をご参照ください。

--investigation

次のオプションは hcp コマンドの説明をご参照ください。

--hpf-p, --wss, --user, --password, --config-file, --log-file, --stat-log-file, --udp, --ws, --wss-no-check-certificate, --ws-proxy, --hpf-p-cong, --hpf-p-mss, --hpf-p-sndbuf, --hpf-p-rcvbuf, --hcp-out, --multi-run

1.10.1 f, force

=====
対応 OS : Linux / Windows

書式 : -f | --force

=====

リンクが既に存在する場合に上書きします（確認を行いません）。

--

例：

```
[user@localhost ~]$ hln -f ...
```

--

1.10.2 i, interactive

=====
対応 OS : Linux / Windows

書式 : -i | --interactive
=====

リンクが既に存在する場合に、削除するか確認を行います。

--

例：

```
[user@localhost ~]$ hln -i ...
```

--

1.10.3 N, no-dereference

=====
対応 OS : Linux / Windows

書式 : -N | --no-dereference
=====

リンクとして指定したパスがディレクトリを指すシンボリックリンクの場合に、このリンクを解決(Dereference)せずに処理します。このオプションを指定しない場合のリンク先のディレクトリにターゲットのファイル名でリンクが作成される動作を変更します。

--

例：

```
[user@localhost ~]$ hln -N ...
```

--

1.10.4 s, symbolic

=====
対応 OS : Linux / Windows

書式 : -s | --symbolic
=====

シンボリックリンクを作成します。

--

例：

```
[user@localhost ~]$ hln -s ...  
--
```

1.10.5 L, logical

```
=====
対応 OS : Linux / Windows
書式 : -L | --logical
=====
```

ハードリンクを行う場合に、ターゲットのパスを論理名として扱います。パスがシンボリックリンクの場合は、解決したパスを使用します。

```
--
例 :
[user@localhost ~]$ hln -L ...
--
```

1.10.6 P, physical

```
=====
対応 OS : Linux / Windows
書式 : -P | --physical
=====
```

ハードリンクを行う場合に、ターゲットのパスを物理名として扱います。パスがシンボリックリンクの場合は、リンク先を解決せずに処理します。

```
--
例 :
[user@localhost ~]$ hln -P ...
--
```

1.10.7 V, version

```
=====
対応 OS : Linux / Windows
書式 : -V | --version
=====
```

hln コマンドのバージョンを表示します。

```
--
例 :
[user@localhost ~]$ hln -V
hln client (hln) 1.3.0_42 / Linux (HpFP2 2.0.0.91_16)
--
```

1.10.8 config-test

=====
対応 OS : Linux / Windows

書式 : --config-test
=====

hln コマンドの入力パラメータ及び設定情報を出力します。

--

例 :

[user@localhost ~]\$ hln -t

...

Number of Logical Processors : 2

Number of Physical Processors : 2

Command parameters

force : disable

interactive : disable

no-dereference : disable

symbolic : disable

logical : disable

physical : enable

host : disable

port : disable

hpfp : disable (TCP used)

ws : disable (TCP used)

ws-proxy : -

wss-no-check-certificate : disable

version : disable

config-test : enable

help : disable

user : disable

password : disable

config-file : disable

log-file : disable

stat-log-file : disable

hcp-out : - [.hcp.out]

multi-run : disable

Target source:

Target destination:

Configuration parameters

PubkeyAuthentication : yes

HCP tools コマンドリファレンス

```
WinLogonUserAuthentication      : yes
PAMAuthentication              : yes
LocalPasswordAuthentication    : yes
RequireServerCertificateSecurity : yes
RejectFallbackServerCertificateSecurity : yes
IgnoreCertificateCNInvalid      : no
IgnoreCertificateDateInvalid    : no
IgnoreUnknownCA                 : no
IgnoreRevocation                : no
WSSIgnoreCertificateCNInvalid   : no
WSSIgnoreCertificateDateInvalid : no
WSSIgnoreUnknownCA             : no
WSSIgnoreRevocation             : no
WSSOptions                     : -
WSSCipherSuites                 : -
WSSCipherList                   : -
CompressLevel                   : -1
HeaderCompress                  : yes
ContentCompress                 : yes
OCSPRevocationEnabled           : yes
StrictHostKeyChecking           : ask
ApplicationStatLog               : yes Rotation[size=no:0:0, pattern=no:]
o:]
TransportStatLog                : no Rotation[size=no:0:0, pattern=no:]
PrivateKeySearchDir              : - [/etc/hcp/keys]
PrivateKeyFile                   : - [~/.hcp/id_rsa]
CACertificateFile                : - [/etc/hcp/cacert.pem]
CACertificatePath                : -
CARevocationFile                : - [/etc/hcp/crl.pem]
CARevocationPath                : -
WSSCACertificateFile            : -
ProtocolVersion                 : 2
MaxConcurrentThread              : 0
UDPTransportExtensionBufferSize : 2147483648
TCPTransportSocketSendBuffer    : 0
MaxRequestFileEntryAtOnce       : 50
MaxReceiveFileSize              : unlimited
MaxSendFileSize                 : unlimited
MaxBufferSize                   : 1073741824
MaxReceiveRate                  : 10000000000
MaxSendRate                     : 10000000000
InitHeaderBlockSize             : 51200
InitContentBlockSize            : 1048576
MaxHeaderBlockSize              : 51200
MaxContentBlockSize             : 1048576
TransportTimeout                : 180
FileLock                        : no
```

```

FileLockTrials                : 0
FileLockTrialInterval         : 3
AcceptableCryptMethod         : AES256/CTR/VMAC AES256/CBC AES128/CBC
[Intel:AES-NI=yes]
AcceptableDigestMethod        : MM128 SHA256 SHA160
DisableDataIntegrityChecking  : no
AcceptDataIntegrityCheckingOnRejection : yes
TransportCharEncoding         : UTF8
HostEncoding                   : UTF8
ApplicationLog                 : INFO no Rotation[size=no:0:0, pattern
=no:]
PubkeyAuthenticationPrior     : yes
ApplicationStatLogSecurityEx   : yes
--

```

1.10.9 h, help

hln コマンドのヘルプを表示します。

1.10.10 host

```

=====
対応 OS : Linux / Windows
書式 : --host=<remote-host>
-----
remote-host
既定値 : なし
値の範囲 : IP アドレス もしくは ホスト名
=====

```

接続先のリモートホストを指定します。リンクターゲット、リンク名もしくはディレクトリを指定するパスに適用されます。このパラメータを指定した場合、このパスのホスト名を省略することができます。

```

--
例 :
[user@localhost ~]$ hln --host=192.168.100.100 ...
--

```

1.10.11 port

```

=====
対応 OS : Linux / Windows
書式 : --port=<remote-port>
-----
remote-port
既定値 : なし

```

値の範囲 : ポート番号

=====
接続先のリモートホストのサービスポート番号を指定します。リンクターゲット、
リンク名もしくはディレクトリを指定するパスに適用されます。

--
例 :
[user@localhost ~]\$ hln --port=1874 ...
--

1.10.12 hcp-out

=====
対応 OS : Linux / Windows
書式 : --hcp-out=<output-path>

output-path
既定値 : なし
値の範囲 : ファイルシステムのパス文字列
=====

リンク作成の実行記録を出力するファイルを指定します。

--
例 :
[user@localhost ~]\$ hln -v ...
TARGET0 127.0.0.1:1874:/home/user/Desktop/hcp_hln01.txt
TARGET1 127.0.0.1:1874:/home/user/Desktop/hcp_hln02.txt
LINK_NAME/DIR 127.0.0.1:1874:/home/user/Desktop/hcp_hln_dir
OK 0000 FL 00000001 /home/user/Desktop/hcp_hln01.txt
OK 0000 FL 00000002 /home/user/Desktop/hcp_hln02.txt
EXIT 0 REASON 0000
[user@localhost ~]\$
--

--
例 :
[user@localhost ~]\$ hln -v ...
--

1.11 hchmod

Usage: hchmod [OPTION]... MODE[,MODE...] [USER@]HOST[:PORT]:FILE [:FILE
E]...
or: hchmod [OPTION]... OCTAL-MODE [USER@]HOST[:PORT]:FILE [:FILE]...
or: hchmod [OPTION]... --host=HOST [--port=PORT] [--user=USER] MODE[,M
ODE...] FILE...

or: hchmod [OPTION]... --host=HOST [--port=PORT] [--user=USER] OCTAL-MODE FILE...

hchmod コマンドは、リモート（サーバ、hcpd サーバ）上でファイルのパーミッションを変更するコマンドです。本コマンドは、Windows サービスではサポートしません。

短縮名	オプション名	概略
v	verbose	実行記録を標準出力に出力する
	hpfp	HpFP プロトコルを使用する
	wss	WebSocket プロトコル(SSL/TLS)を使用する
V	version	バージョンを表示する
	config-test	入力パラメータ及び設定ファイルのロード結果を表示する
h	help	ヘルプを表示する
	user	ユーザ名
	password	パスワード
	host	リモートホスト指定
	port	リモートポート指定
	config-file	設定ファイル
	log-file	ログファイル
	stat-log-file	統計ログファイル
	udp	UDP (HpFP)プロトコルを使用する（ポート分離型）
	ws	WebSocket プロトコル（平文通信）を使用する
	wss-no-check-certificate	WebSocket SSL/TLS サーバ証明書検証無効化
	ws-proxy	WebSocket プロキシサーバ指定
	hpfp-cong	HpFP プロトコルの輻輳制御モードを指定
	hpfp-mss	HpFP プロトコルの MSS 指定
	hpfp-sndbuf	HpFP プロトコルの送信バッファサイズ指定
	hpfp-rcvbuf	HpFP プロトコルの受信バッファサイズ指定

	hcp-out	実行記録を出力するファイルを指定する
	investigation	調査モード
	multi-run	多重起動モード

次のオプションは hcpd コマンドの説明をご参照ください。

--investigation

次のオプションは hcp コマンドの説明をご参照ください。

--hpfp, --wss, --user, --password, --config-file, --log-file, --stat-log-file, --udp, --ws, --wss-no-check-certificate, --ws-proxy, --hpfp-cong, --hpfp-mss, --hpfp-sndbuf, --hpfp-rcvbuf, --hcp-out, --multi-run

1.11.1 R, recursive

対応 OS : Linux / Windows

書式 : -R | --recursive

ディレクトリを再帰的に探索してパーミッションの変更を適用します。

--

例 :

```
[user@localhost ~]$ hchmod -R ...
```

--

1.11.2 V, version

対応 OS : Linux / Windows

書式 : -V | --version

hchmod コマンドのバージョンを表示します。

--

例 :

```
[user@localhost ~]$ hchmod -V
hchmod client (hchmod) 1.3.0_42 / Linux (HpFP2 2.0.0.91_16)
```

--

1.11.3 config-test

対応 OS : Linux / Windows

書式 : --config-test

hchmod コマンドの入力パラメータ及び設定情報を出力します。

--

例 :

```
[user@localhost ~]$ hchmod --config-test
```

...

Number of Logical Processors : 2

Number of Physical Processors : 2

Command parameters

```
modes      : -
recursive  : disable
host       : disable
port       : disable
hpfp       : disable (TCP used)
ws         : disable (TCP used)
ws-proxy   : -
wss-no-check-certificate : disable
version    : disable
config-test : enable
help       : disable
user       : disable
password   : disable
config-file : disable
log-file   : disable
stat-log-file : disable
hcp-out    : - [.hcp.out]
multi-run  : disable
```

Configuration parameters

```
PubkeyAuthentication      : yes
WinLogonUserAuthentication : yes
PAMAuthentication        : yes
LocalPasswordAuthentication : yes
RequireServerCertificateSecurity : yes
RejectFallbackServerCertificateSecurity : yes
IgnoreCertificateCNInvalid : no
IgnoreCertificateDateInvalid : no
IgnoreUnknownCA           : no
IgnoreRevocation           : no
WSSIgnoreCertificateCNInvalid : no
WSSIgnoreCertificateDateInvalid : no
WSSIgnoreUnknownCA        : no
WSSIgnoreRevocation        : no
```

HCP tools コマンドリファレンス

```

WSSOptions : -
WSSCipherSuites : -
WSSCipherList : -
CompressLevel : -1
HeaderCompress : yes
ContentCompress : yes
OCSPRevocationEnabled : yes
StrictHostKeyChecking : ask
ApplicationStatLog : yes Rotation[size=no:0:0, pattern=no:]
o:]
TransportStatLog : no Rotation[size=no:0:0, pattern=no:]
PrivateKeySearchDir : - [/etc/hcp/keys]
PrivateKeyFile : - [~/.hcp/id_rsa]
CACertificateFile : - [/etc/hcp/cacert.pem]
CACertificatePath : -
CARevocationFile : - [/etc/hcp/crl.pem]
CARevocationPath : -
WSSCACertificateFile : -
ProtocolVersion : 2
MaxConcurrentThread : 0
UDPTransportExtensionBufferSize : 2147483648
TCPTransportSocketSendBuffer : 0
MaxRequestFileEntryAtOnce : 50
MaxReceiveFileSize : unlimited
MaxSendFileSize : unlimited
MaxBufferSize : 1073741824
MaxReceiveRate : 10000000000
MaxSendRate : 10000000000
InitHeaderBlockSize : 51200
InitContentBlockSize : 1048576
MaxHeaderBlockSize : 51200
MaxContentBlockSize : 1048576
TransportTimeout : 180
FileLock : no
FileLockTrials : 0
FileLockTrialInterval : 3
AcceptableCryptMethod : AES256/CTR/VMAC AES256/CBC AES128/CBC
[Intel:AES-NI=yes]
AcceptableDigestMethod : MM128 SHA256 SHA160
DisableDataIntegrityChecking : no
AcceptDataIntegrityCheckingOnRejection : yes
TransportCharEncoding : UTF8
HostEncoding : UTF8
ApplicationLog : INFO no Rotation[size=no:0:0, pattern=no:]
=no:]
PubkeyAuthenticationPrior : yes

```

```
ApplicationStatLogSecurityEx      : yes
```

```
--
```

1.11.4 h, help

```
=====
対応 OS : Linux / Windows
```

```
書式 : -h | --help
=====
```

hchmod コマンドのヘルプを表示します。

```
--
```

例 :

```
[user@localhost ~]$ hchmod -h
```

```
--
```

1.11.5 host

```
=====
対応 OS : Linux / Windows
```

```
書式 : --host=<remote-host>
-----
```

remote-host

既定値 : なし

値の範囲 : IP アドレス もしくは ホスト名

```
=====
```

接続先のリモートホストを指定します。リンクターゲット、リンク名もしくはディレクトリを指定するパスに適用されます。このパラメータを指定した場合、このパスのホスト名を省略することができます。

```
--
```

例 :

```
[user@localhost ~]$ hchmod --remote-host=192.168.100.100 ...
```

```
--
```

1.11.6 port

```
=====
対応 OS : Linux / Windows
```

```
書式 : --port=<remote-port>
-----
```

remote-port

既定値 : なし

値の範囲 : ポート番号

```
=====
```


接続先のリモートホストのサービスポート番号を指定します。リンクターゲット、リンク名もしくはディレクトリを指定するパスに適用されます。

```
--
例 :
[user@localhost ~]$ hchmod --remote-port=1874 ...
--
```

1.11.7 hcp-out

```
=====
対応 OS : Linux / Windows
書式 : --hcp-out=<output-path>
-----
```

```
output-path
既定値 : なし
値の範囲 : ファイルシステムのパス文字列
=====
```

パーミッション変更の実行記録を出力するファイルを指定します。

```
--
例 :
[user@localhost ~]$ hchmod -v ...
TODO
EXIT 0 REASON 0000
[user@localhost ~]$
--
```

```
--
例 :
[user@localhost ~]$ hchmod -v ...
--
```

1.12 hchown

```
Usage: hchown [OPTION]... OWNER[:[GROUP]] [USER@]HOST[:PORT]:FILE [:FILE
E]...
    or: hchown [OPTION]... --host=HOST [--port=PORT] [--user=USER] OWNER[:
[GROUP]] FILE...
```

hchown コマンドは、リモート（サーバ、hcpd サーバ）上でファイルの所有者またはグループを変更するコマンドです。本コマンドは、**Windows** サービスではサポートしません。

短縮名	オプション名	概略
-----	--------	----

d	no-dereference	シンボリックリンク先ディレクトリの解決を無効にする
R	recursive	再帰的にディレクトリを探索して変更
s	follow-cmd-link-dir	コマンド引数のディレクトリリンク探索設定
S	follow-all	ディレクトリへのシンボリックリンクをたどる
D	no-follow	シンボリックリンクはたどらない
	hpfp	HpFP プロトコルを使用する
	wss	WebSocket プロトコル(SSL/TLS)を使用する
V	version	バージョンを表示する
	config-test	入力パラメータ及び設定ファイルのロード結果を表示する
h	help	ヘルプを表示する
	user	ユーザ名
	password	パスワード
	host	リモートホスト指定
	port	リモートポート指定
	config-file	設定ファイル
	log-file	ログファイル
	stat-log-file	統計ログファイル
	udp	UDP (HpFP)プロトコルを使用する（ポート分離型）
	ws	WebSocket プロトコル（平文通信）を使用する
	wss-no-check-certificate	WebSocket SSL/TLS サーバ証明書検証無効化
	ws-proxy	WebSocket プロキシサーバ指定
	hpfp-cong	HpFP プロトコルの輻輳制御モードを指定
	hpfp-mss	HpFP プロトコルの MSS 指定
	hpfp-sndbuf	HpFP プロトコルの送信バッファサイズ指定
	hpfp-rcvbuf	HpFP プロトコルの受信バッファサイズ指定
	hcp-out	実行記録を出力するファイルを指定する
	investigation	調査モード

	multi-run	多重起動モード
--	-----------	---------

次のオプションは `hcpd` コマンドの説明をご参照ください。

`--investigation`

次のオプションは `hcp` コマンドの説明をご参照ください。

`--hpfp`, `--wss`, `--user`, `--password`, `--config-file`, `--log-file`, `--stat-log-file`, `--udp`, `--ws`, `--wss-no-check-certificate`, `--ws-proxy`, `--hpfp-cong`, `-hpfp-mss`, `--hpfp-sndbuf`, `--hpfp-rcvbuf`, `--hcp-out`, `--multi-run`

1.12.1 d, no-dereference

=====
対応 OS : Linux / Windows

書式 : `-d` | `--no-dereference`
=====

シンボリックリンクを解決せずにシンボリックリンクに変更を適用します。

--

例 :

`[user@localhost ~]$ hchown -d ...`

--

1.12.2 R, recursive

=====
対応 OS : Linux / Windows

書式 : `-R` | `--recursive`
=====

ディレクトリを再帰的に探索して変更を行います。

--

例 :

`[user@localhost ~]$ hchown -R ...`

--

1.12.3 s, follow-cmd-link-dir

=====
対応 OS : Linux / Windows

書式 : `-s` | `--follow-cmd-link-dir`
=====

コマンドラインの引数のシンボリックリンクがディレクトリへのリンクであった場合は、そのディレクトリをたどって変更を行います。

--

例 :

```
[user@localhost ~]$ hchown -s ...
```

--

1.12.4 S, follow-all

=====

対応 OS : Linux / Windows

書式 : -S | --follow-all

=====

ディレクトリに解決されるシンボリックリンクはそのディレクトリをたどって変更を行います。

--

例 :

```
[user@localhost ~]$ hchown -S ...
```

--

1.12.5 D, no-follow

=====

対応 OS : Linux / Windows

書式 : -D | --no-follow

=====

シンボリックリンクをたどらずに変更を行います。

--

例 :

```
[user@localhost ~]$ hchown -D ...
```

--

1.12.6 V, version

=====

対応 OS : Linux / Windows

書式 : -V | --version

=====

hchown コマンドのバージョンを表示します。

--

例 :

```
[user@localhost ~]$ hchown -V
hchown client (hchown) 1.3.0_42 / Linux (HpFP2 2.0.0.91_16)
```

--

1.12.7 config-test

=====
対応 OS : Linux / Windows

書式 : --config-test
=====

hchown コマンドの入力パラメータ及び設定情報を出力します。

--

例 :

```
[user@localhost ~]$ hchmod --config-test
```

...

Number of Logical Processors : 2

Number of Physical Processors : 2

Command parameters

```
no-dereference      : disable
recursive           : disable
follow-cmd-link-dir  : disable
follow-all          : disable
no-follow            : enable
host                 : disable
port                 : disable
hpfp                 : disable (TCP used)
ws                   : disable (TCP used)
ws-proxy             : -
wss-no-check-certificate : disable
version              : disable
config-test          : enable
help                 : disable
user                 : disable
password             : disable
config-file          : disable
log-file             : disable
stat-log-file        : disable
hcp-out              : - [.hcp.out]
multi-run            : disable
```

Configuration parameters

```
PubkeyAuthentication      : yes
WinLogonUserAuthentication : yes
PAMAuthentication         : yes
LocalPasswordAuthentication : yes
RequireServerCertificateSecurity : yes
```

```

RejectFallbackServerCertificateSecurity : yes
IgnoreCertificateCNInvalid              : no
IgnoreCertificateDateInvalid            : no
IgnoreUnknownCA                        : no
IgnoreRevocation                       : no
WSSIgnoreCertificateCNInvalid           : no
WSSIgnoreCertificateDateInvalid         : no
WSSIgnoreUnknownCA                     : no
WSSIgnoreRevocation                     : no
WSSOptions                             : -
WSSCipherSuites                        : -
WSSCipherList                          : -
CompressLevel                          : -1
HeaderCompress                         : yes
ContentCompress                        : yes
OCSPRevocationEnabled                   : yes
StrictHostKeyChecking                   : ask
ApplicationStatLog                      : yes Rotation[size=no:0:0, pattern=no:]
o:]
TransportStatLog                       : no Rotation[size=no:0:0, pattern=no:]
PrivateKeySearchDir                     : - [/etc/hcp/keys]
PrivateKeyFile                          : - [~/.hcp/id_rsa]
CACertificateFile                       : - [/etc/hcp/cacert.pem]
CACertificatePath                       : -
CARevocationFile                       : - [/etc/hcp/crl.pem]
CARevocationPath                       : -
WSSCACertificateFile                   : -
ProtocolVersion                        : 2
MaxConcurrentThread                     : 0
UDPTransportExtensionBufferSize         : 2147483648
TCPTransportSocketSendBuffer            : 0
MaxRequestFileEntryAtOnce               : 50
MaxReceiveFileSize                      : unlimited
MaxSendFileSize                         : unlimited
MaxBufferSize                          : 1073741824
MaxReceiveRate                          : 10000000000
MaxSendRate                            : 10000000000
InitHeaderBlockSize                     : 51200
InitContentBlockSize                    : 1048576
MaxHeaderBlockSize                      : 51200
MaxContentBlockSize                     : 1048576
TransportTimeout                        : 180
FileLock                               : no
FileLockTrials                          : 0
FileLockTrialInterval                   : 3
AcceptableCryptMethod                   : AES256/CTR/VMAC AES256/CBC AES128/CBC
[Intel:AES-NI=yes]

```

```
AcceptableDigestMethod      : MM128 SHA256 SHA160
DisableDataIntegrityChecking : no
AcceptDataIntegrityCheckingOnRejection : yes
TransportCharEncoding       : UTF8
HostEncoding                 : UTF8
ApplicationLog               : INFO no Rotation[size=no:0:0, pattern
=no:]
PubkeyAuthenticationPrior    : yes
ApplicationStatLogSecurityEx : yes
--
```

1.12.8 h, help

```
=====
対応 OS : Linux / Windows
書式 : -h | --help
=====
```

hchown コマンドのヘルプを表示します。

```
--
例 :
[user@localhost ~]$ hchown -h
--
```

1.12.9 host

```
=====
対応 OS : Linux / Windows
書式 : --host=<remote-host>
-----
```

```
remote-host
既定値 : なし
値の範囲 : IP アドレス もしくは ホスト名
=====
```

接続先のリモートホストを指定します。リンクターゲット、リンク名もしくはディレクトリを指定するパスに適用されます。このパラメータを指定した場合、このパスのホスト名を省略することができます。

```
--
例 :
[user@localhost ~]$ hchown --remote-host=192.168.100.100 ...
--
```

1.12.10 port

```
=====
対応 OS : Linux / Windows
```

書式 : `--port=<remote-port>`

`remote-port`

既定値 : なし

値の範囲 : ポート番号

接続先のリモートホストのサービスポート番号を指定します。リンクターゲット、リンク名もしくはディレクトリを指定するパスに適用されます。

--

例 :

```
[user@localhost ~]$ hchown --remote-port=1874 ...
```

--

1.12.11 hcp-out

対応 OS : Linux / Windows

書式 : `--hcp-out=<output-path>`

`output-path`

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

所有者とグループ変更の実行記録を出力するファイルを指定します。

--

例 :

```
[user@localhost ~]$ hchown -v ...
```

```
TODO
```

```
EXIT 0 REASON 0000
```

```
[user@localhost ~]$
```

--

--

例 :

```
[user@localhost ~]$ hchown -v ...
```

--

2 設定リファレンス

2.1 hcpd.conf

hcpd デーモンの設定を記述します。

項目名	説明
ProtocolVersion	プロトコルバージョン(2 固定)
TCPListenAddress	TCP サービス待機アドレス設定
TCPServiceSocketSendBuffer	TCP 送信バッファ設定
HPFPListenAddress	HpFP サービス待機アドレス設定
UDPListenAddress	HpFP ポート分離型サービス待機アドレス設定
UDPServiceExtensionBufferSize	HpFP サービス拡張バッファサイズ設定
WSSListenAddress	WebSocket(SSL/TLS)サービス待機アドレス設定
WSSOptions	WSS オプション設定
WSSCipherSuites	WSS TLS 1.3 Cipher Suites オプション
WSSCipherList	WSS TLS 1.2 or earlier Cipher Suites オプション
WSListenAddress	WebSocket（平文通信）サービス待機アドレス設定
ListenServiceBonding	サービスボンディング設定
UseServerCertificateSecurity	サーバ証明書セキュリティ機能を使用する
RequireServerCertificateSecurity	サーバ証明書セキュリティ機能を要求する
ServerKeyFile	サーバ鍵ファイル（".pub"ファイルが存在する場合は、秘密鍵と公開鍵のペアにより通信セキュリティを確保します）
ServerCertificateFile	サーバ証明書ファイル
ServerCertificateChainFile	サーバ証明書証明書連鎖ファイル
LocalPasswordAuthentication	LPA 認証
PAMAuthentication	PAM 認証
PubkeyAuthentication	RSA 認証

WinLogonUserAuthentication	Windows 認証
PerformSystemAuthenticationRegardlessUsers	システム認証実行制御
UserDirectoryFallbackAvailable	ユーザディレクトリ後退制御
RejectOnUserHomeDirectoryNotFound	ユーザホームディレクトリ検査
UsePrivilegeSeparation	特権分離
PrivilegeSeparationMinimumUID	特権分離最小 UID
PrivilegeSeparationMinimumGID	特権分離最小 GID
PrivilegeSeparationUser	特権分離時デフォルトユーザ
PrivilegeSeparationUmask	特権分離時 Umask
PrivilegeSeparationUmaskAnonymous	特権分離時 Umask(匿名ユーザ)
ApplyUserPermission	ユーザパーミッション適用
AllowUsers	アクセス許可ユーザ設定
AllowGroups	アクセス許可グループ設定
DenyUsers	アクセス拒否ユーザ設定
DenyGroups	アクセス拒否グループ設定
AuthorizedKeySearchDir	認証鍵探索ディレクトリ
AuthorizedKeyFile	認証鍵ファイル
CACertificateFile	CA 証明書ファイル
CACertificatePath	CA 証明書探索ディレクトリ (予約)
CARevocationFile	CA CRL ファイル
CARevocationPath	CA CRL 探索ディレクトリ (予約)
OCSPRevocationEnabled	OCSP (オンライン証明書失効検査) 機能
LocalUserFile	ユーザ設定ファイル
LocalPasswordFile	LPA 認証資格情報設定ファイル
AcceptableCryptMethod	暗号方式
AcceptableDigestMethod	ダイジェスト方式
RequireDataIntegrityChecking	暗号通信メッセージ検査要求
TransportCharEncoding	トランスポート文字エンコード

HostEncoding	ホスト文字エンコード
HeaderCompress	ヘッダ圧縮（予約）
ContentCompress	コンテンツ圧縮（予約）
MaxConcurrentThread	使用スレッド数制限(Linux)
MaxTotalConnection	接続数制限
MaxTcpConnection	TCP 接続数制限
MaxUdpConnection	UDP (HpFP)接続数制限
MaxWsConnection	WebSocket 接続数制限
MaxConnectionPerUser	ユーザ当たりの接続数制限
MaxConnectionPerSec	秒間接続数制限
MaxReceiveFileSize	受信ファイルサイズ制限
MaxSendFileSize	送信ファイルサイズ制限
MaxRequestFileEntryAtOnce	一括ファイル要求制限
MaxTotalBufferSize	ペイロードバッファメモリ割当制限（システム全体）
MaxBufferSizePerConnection	ペイロードバッファメモリ割当制限（接続単位）
MaxTotalReceiveRate	受信レート制限（システム全体）
MaxTotalSendRate	送信レート制限（システム全体）
MaxReceiveRatePerConnection	受信レート制限（接続単位）
MaxSendRatePerConnection	送信レート制限（接続単位）
InitHeaderBlockSize	初期ヘッダブロックサイズ
InitContentBlockSize	初期コンテンツブロックサイズ
MaxHeaderBlockSize	最大ヘッダブロックサイズ
MaxContentBlockSize	最大コンテンツブロックサイズ
FileLock	ファイルロック
FileLockTrials	ファイルロック再試行回数
FileLockTrialInterval	ファイルロック試行間隔（秒）
AtomicLikeSaving	アトミック（不可分）ファイル保存
AtomicLikeSavingThreshold	アトミック（不可分）ファイル保

	存 実行閾値
AtomicLikeSavingRejectOverwriteRequest	アトミック（不可分）ファイル保存一時ファイル上書き要求拒否
TransportTimeout	トランスポートタイムアウト
IdleTimeout	アイドルタイムアウト
DocPoint	ドキュメントポイント（ファイルシステムアクセス制限）
AccessList	アクセス制御リスト
SyslogOption	syslog オプション
SyslogFacility	syslog ファシリティ
SystemLog	ログ設定
SystemLogLevel	ログレベル
ApplicationStatLog	アプリケーション統計
TransportStatLog	トランスポート統計
SystemStatLog	システム統計
FileOperationLog	ファイル操作ログ
CallbackScript	アプリ連携コールバック
MemoryTransferConcurrency	メモリ転送並列設定
MaxReadRatePerConnection	読込レート制限（接続単位）
MaxWriteRatePerConnection	書込レート制限（接続単位）
EnsureDestinationInFileTransfer	宛先ディレクトリ保証制御
StatLogPerUserInPrivilegeSeparation	統計ログ出力先制御
ApplicationStatLogSecurityEx	アプリ統計ログセキュリティ詳細情報出力設定

2.1.1 TCPListenAddress

```
=====
対応 OS : Linux.x86 / Windows
書式 : TCPListenAddress <tcp_service_addr>[:<tcp_service_port>[:<mcd>]][
<acl_name>]
-----
tcp_service_addr
既定値 : なし
値の範囲 : IP アドレス
```

tcp_service_port

書式 : <decimal_number>

既定値 : 874

値の範囲 : decimal_number は、1 - 65535 の範囲

mcd

既定値 : 物理コア数 + 1, 論理コア数/2 + 1, 25 or 1

値の範囲 : 1 - 65535

acl_name

既定値 : なし

値の範囲 : 設定ファイル中に記述されたアクセスコントロールリストの名前
=====

クライアントからの接続を受け付ける TCP サービスを定義します。

tcp_service_addr は、TCP サービスの IP アドレスを指定します。

tcp_service_port は、同サービスのポート番号を指定します。

mcd は、オプションです。このサービスでクライアントに提供するセッションあたりの接続の数を指定します。1 を指定した場合は、セッションは単一の接続で動作します。2 以上を指定した場合は、接続の受付とセッションのネゴシエーションが完了した後に、指定された数から 1 を差し引いた数だけ追加で接続が行われ並列で通信を行います。指定しない場合は、CPU の物理コア数、論理コア数及びライセンスから次の何れかの値に決まります。

- 物理コア数 + 1 (物理コア数 < 論理コア数 かつ 物理コア数 > 1 の場合。但し、最大 25)

- 論理コア数/2 + 1 (物理コア数 = 論理コア数 かつ 論理コア数/2 > 1 の場合。但し、最大 25)

- 1 (物理コア数もしくは論理コア数/2 が 1 の場合やライセンスに多重化機能が含まれない場合)

acl_name は、オプションでこの TCP サービスに設定するアクセスコントロールリストを名前指定します。このオプションを省略した場合、もしくは指定された名前がいずれのアクセスコントロールリストの名前と一致しない場合は、無名のアクセスコントロールリストがサービスに設定されます。

--

例 :

TCPListenAddress 0.0.0.0:1874

--

2.1.2 TCPServiceSocketSendBuffer

対応 OS : Linux.x86 / Windows

書式 : TCPServiceSocketSendBuffer <snd-buf-size>

snd-buf-size

書式 : <decimal_number>[[(T|G|M|K)]B]

既定値 : 0

値の範囲 : 符号なし倍長整数 (バイト単位)

=====

TCP 通信で指定する送信バッファのサイズ (バイト単位) を設定します。0 を指定すると未指定となります。

100G 環境などで TCP 通信性能のチューニングが必要な場合などに使用します。
通常は変更する必要はありません。

--

例 :

TCPServiceSocketSendBuffer 128MB

--

2.1.3 HPFPListenAddress

対応 OS : Linux.x86 / Windows

書式 : HPFPListenAddress <hpf_service_addr>[:<hpf_service_port>[:<hpf_sndbuf>[:<hpf_rcvbuf>[:<hpf_mss>[:<mcd>]]]]][<acl_name>]

hpf_service_addr

既定値 : なし

値の範囲 : IP アドレス値

hpf_service_port

書式 : <decimal_number>

既定値 : 65520

値の範囲 : decimal_number は、1 - 65535 の範囲

hpf_sndbuf

書式 : (DEFAULT | D | <decimal_number>[[(T|G|M|K)]B])

既定値 : 100MB

値の範囲 : バイト換算で符号なし倍長整数の範囲。 D は DEFAULT の略記。

hpfp_rcvbuf

書式 : (DEFAULT | D | <decimal_number>[[T|G|M|K)]B])

既定値 : 200MB

値の範囲 : バイト換算で符号なし倍長整数の範囲。 D は DEFAULT の略記。

hpfp_mss

書式 : (DEFAULT | D | NONE | N | <decimal_number>[[T|G|M|K)]B])

既定値 : NONE

値の範囲 : バイト換算で符号なし整数の範囲。 D は DEFAULT の略記。 N は NONE の略記。

mcd

既定値 : 物理コア数 + 1, 論理コア数/2 + 1, 25 or 1

値の範囲 : 1 - 65535

acl_name

既定値 : なし

値の範囲 : 設定ファイル中に記述されたアクセスコントロールリストの名前

=====

クライアントからの接続を受け付ける HpFP サービスを定義します。

hpfp_service_addr は、HpFP サービスの IP アドレスを指定します。

hpfp_service_port は、オプションです。HpFP プロトコルで使用する UDP トランスポートのポート番号を指定します。

hpfp_sndbuf は、オプションです。HpFP プロトコルで使用する送信バッファのサイズを指定します。D を指定すると既定値を使用します。

hpfp_rcvbuf は、オプションです。HpFP プロトコルで使用する受信バッファのサイズを指定します。D を指定すると既定値を使用します。

hpfp_mss は、オプションです。HpFP プロトコルで使用する MSS を指定します。D を指定すると既定値を使用します。N を指定した場合、HpFP プロトコルによる MTU 探索を行い値を決定します。

mcd は、オプションです。このサービスでクライアントに提供するセッションあたりの接続の数を指定します。1 を指定した場合は、セッションは単一の接続で動作します。2 以上を指定した場合は、接続の受付とセッションのネゴシエーションが完了した後に、指定された数から 1 を差し引いた数だけ追加で接続が行われ並列で通信を行います。指定しない場合は、CPU の物理コア数、論理コア数及びライセンスから次の何れかの値に決まります。

- 物理コア数 + 1 (物理コア数 < 論理コア数 かつ 物理コア数 > 1 の場合。但し、最大 25)
- 論理コア数/2 + 1 (物理コア数 = 論理コア数 かつ 論理コア数/2 > 1 の場合。但し、最大 25)
- 1 (物理コア数もしくは論理コア数/2 が 1 の場合やライセンスに多重化機能が含まれない場合)

acl_name は、オプションです。TCP サービと同様。

```
--  
例 :  
HPFPListenAddress 0.0.0.0:10000  
--
```

2.1.4 UDPListenAddress

```
=====  
対応 OS : Linux.x86 / Windows  
書式 : UDPListenAddress <hpfp_service_addr>[:<hpfp_service_port>[:<hpfp_u  
dp_port>[:<hpfp_sndbuf>[:<hpfp_rcvbuf>[:<hpfp_mss>[:<mcd>]]]]][ <acl_nam  
e>]
```

hpfp_service_addr

既定値 : なし

値の範囲 : IP アドレス値

hpfp_service_port

書式 : <decimal_number>

既定値 : 884

値の範囲 : decimal_number は、1 - 65535 の範囲

hpfp_udp_port

書式 : (DEFAULT | D | <decimal_number>)

既定値 : 65520

値の範囲 : decimal_number は、1 - 65535 の範囲。D は DEFAULT の略記。

hpfp_sndbuf

書式 : (DEFAULT | D | <decimal_number>[[T|G|M|K)]B])

既定値 : 100MB

値の範囲 : バイト換算で符号なし倍長整数の範囲。D は DEFAULT の略記。

hpfp_rcvbuf

書式 : (DEFAULT | D | <decimal_number>[[T|G|M|K)]B])

既定値 : 200MB

値の範囲 : バイト換算で符号なし倍長整数の範囲。 D は DEFAULT の略記。

hpfp_mss

書式 : (DEFAULT | D | NONE | N | <decimal_number>[[(T|G|M|K)]B])

既定値 : NONE

値の範囲 : バイト換算で符号なし整数の範囲。 D は DEFAULT の略記。 N は NONE の略記。

mcd

既定値 : 物理コア数 + 1, 論理コア数/2 + 1, 25 or 1

値の範囲 : 1 - 65535

acl_name

既定値 : なし

値の範囲 : 設定ファイル中に記述されたアクセスコントロールリストの名前

=====

本設定項目は、廃止予定(deprecated)です。

クライアントからの接続を受け付ける HpFP(UDP)サービスを定義します。この設定項目は、サービスのポート番号と UDP トランスポート番号を分離する方式の HpFP サービスを定義する方法を提供します。

hpfp_service_addr は、HpFP サービスの IP アドレスを指定します。

hpfp_service_port はオプションです。サービスのポート番号を指定します。

hpfp_udp_port は、オプションです。HpFP プロトコルで使用する UDP トランスポートのポート番号を指定します。D を指定すると既定値を使用します。

hpfp_sndbuf は、オプションです。HpFP プロトコルで使用する送信バッファのサイズを指定します。D を指定すると既定値を使用します。

hpfp_rcvbuf は、オプションです。HpFP プロトコルで使用する受信バッファのサイズを指定します。D を指定すると既定値を使用します。

hpfp_mss は、オプションです。HpFP プロトコルで使用する MSS を指定します。D を指定すると既定値を使用します。N を指定した場合、HpFP プロトコルによる MTU 探索を行い値を決定します。

mcd は、オプションです。このサービスでクライアントに提供するセッションあたりの接続の数を指定します。1 を指定した場合は、セッションは単一の接続で動作します。2 以上を指定した場合は、接続の受付とセッションのネゴシエーションが完了した後に、指定された数から 1 を差し引いた数だけ追加で接続が行

われ並列で通信を行います。指定しない場合は、CPU の物理コア数、論理コア数及びライセンスから次の何れかの値に決まります。

- 物理コア数 + 1 （物理コア数 < 論理コア数 かつ 物理コア数 > 1 の場合。但し、最大 25）
- 論理コア数/2 + 1 （物理コア数 = 論理コア数 かつ 論理コア数/2 > 1 の場合。但し、最大 25）
- 1 （物理コア数もしくは論理コア数/2 が 1 の場合やライセンスに多重化機能が含まれない場合）

`acl_name` は、オプションです。TCP サービと同様。

```
--  
例 :  
UDPListenAddress 0.0.0.0:1884:10000  
--
```

2.1.5 UDPServiceExtensionBufferSize

```
=====
```

対応 OS : Linux.x86 / Windows
書式 : UDPServiceExtensionBufferSize <ext-buf-size>

ext-buf-size
既定値 : 2GB
値の範囲 : 符号なし倍長整数 （バイト単位）
=====

HpFP(UDP)サービスで使用する拡張バッファのサイズを指定します。

HpFP セッションでは、遅延やロス及び通信量の増加などに伴い通信用のバッファを指定されたサイズ（UDPListenAddress の `hpfp_sndbuf` 及び `hpfp_rcvbuf`）に拡張します。この拡張するバッファの合計容量を指定された値で制限します。

0 を指定した場合は、この制限を行いません。

また、バッファの初期サイズ（拡張される前のバッファのサイズ）は 1MB です。

```
--  
例 :  
UDPServiceExtensionBufferSize 4GB  
--
```

2.1.6 WSSListenAddress

```
=====
```

対応 OS : Linux.x86 / Windows

書式 : WSSListenAddress <wss_service_addr>[:<wss_service_port>[:<wss_opt_name>[:<wss_cs_name>[:<wss_clist_name>[:<wss_privkey_name>[:<wss_cert_name>[:<mcd>[:<use_hcpcm>]]]]]]]][<acl_name>]

wss_service_addr

既定値 : なし

値の範囲 : IP アドレス値

wss_service_port

既定値 : 443

値の範囲 : ポート番号

wss_opt_name

書式 : (DEFAULT | D | <opt_name>)

既定値 : 空文字列

値の範囲 : WSSOptions の名前。D は DEFAULT の略記

wss_cs_name

書式 : (DEFAULT | D | <cipher_suites_name>)

既定値 : 空文字列

値の範囲 : WSSCipherSuites の名前。D は DEFAULT の略記

wss_clist_name

書式 : (DEFAULT | D | <cipher_list_name>)

既定値 : 空文字列

値の範囲 : WSSCipherLlist の名前。D は DEFAULT の略記

wss_privkey_name

書式 : (DEFAULT | D | <server_key_file_name>)

既定値 : 空文字列

値の範囲 : ServerKeyFile の名前

wss_cert_name

書式 : (DEFAULT | D | <server_cert_file_name>)

既定値 : 空文字列

値の範囲 : ServerCertificateFile の名前

mcd

既定値 : 物理コア数 + 1, 論理コア数/2 + 1, 25 or 1

値の範囲 : 1 - 65535

use_hcpcm

既定値 : no

値の範囲 : yes, no

acl_name

既定値 : なし

値の範囲 : 設定ファイル中に記述されたアクセスコントロールリストの名前
=====

クライアントからの接続を受け付ける WebSocket サービス(SSL/TLS)を定義します。

wss_service_addr は、WebSocet サービス(SSL/TLS)の IP アドレスを指定します。

wss_service_port は、オプションです。同サービスのポート番号を指定します。
省略した場合は 443 が使用されます。

wss_opt_name は、オプションです。WSSOptions の名前を指定します。D を指定すると既定値を使用します。名前が指定されない場合（空文字列）は、無名の WSSOptions が使用されます。

wss_cs_name は、オプションです。WSSCipherSuites の名前を指定します。D を指定すると既定値を使用します。名前が指定されない場合（空文字列）は、無名の WSSCipherSuites が使用されます。

wss_clist_name は、オプションです。WSSCipherList の名前を指定します。D を指定すると既定値を使用します。名前が指定されない場合（空文字列）は、無名の WSSCipherList が使用されます。

wss_privkey_name は、オプションです。SSL/TLS 通信で使用するサーバ鍵（秘密鍵）を設定した ServerKeyFile の名前を指定します。D を指定すると既定値を使用します。名前が指定されない場合は、無名の ServerKeyFile が使用されます。

wss_cert_name は、オプションです。SSL/TLS 通信で使用するサーバ証明書を設定した ServerCertificateFile の名前を指定します。D を指定すると既定値を使用します。名前が指定されない場合は、無名の ServerCertificateFile が使用されます。

mcd は、オプションです。このサービスでクライアントに提供するセッションあたりの接続の数を指定します。1 を指定した場合は、セッションは単一の接続で動作します。2 以上を指定した場合は、接続の受付とセッションのネゴシエーションが完了した後に、指定された数から 1 を差し引いた数だけ追加で接続が行われ並列で通信を行います。指定しない場合は、CPU の物理コア数、論理コア数及びライセンスから次の何れかの値に決まります。

- 物理コア数 + 1 （物理コア数 < 論理コア数 かつ 物理コア数 > 1 の場合。但し、最大 25）

- 論理コア数/2 + 1 (物理コア数 = 論理コア数 かつ 論理コア数/2 > 1 の場合。但し、最大 25)
- 1 (物理コア数もしくは論理コア数/2 が 1 の場合やライセンスに多重化機能が含まれない場合)

use_hcpcm は、オプションです。SSL/TLS 通信をする際にアプリケーションで処理する暗号化処理 (AcceptableCryptMethod) を適用するか設定します。D を指定すると既定値を使用します。

acl_name は、オプションです。TCP サービと同様。

```
--  
例 :  
WSSListenAddress 0.0.0.0:8443  
--
```

2.1.7 WSSOptions

```
=====
```

対応 OS : Linux.x86 / Windows
書式 : WSSOptions <opt_value>[<opt_name>]

opt_value
書式 : (NONE | <openssl_opt_values>)
既定値 : NONE
値の範囲 : OpenSSL で規定される SSL/TLS オプション名のリスト

opt_name
既定値 : なし
値の範囲 : 文字列
=====

SSL/TLS 通信を実行する際に設定する OpenSSL オプションを指定します。指定するオプションの名前は下記 URL に規定される名前を使用します。

https://www.openssl.org/docs/man1.1.1/man3/SSL_CTX_set_options.html
SSL_CTX_set_options

```
--  
例 :  
WSSOptions SSL_OP_NO_COMPRESSION:SSL_OP_NO_SSLv3  
--
```

2.1.8 WSSCipherSuites

```
=====
```

対応 OS : Linux.x86 / Windows

書式 : WSSCipherSuites <cs_value>[<cs_name>]

cs_value

書式 : (NONE | <openssl_cipher_suite_values>)

既定値 : NONE

値の範囲 : OpenSSL で規定される Cipher Suites パラメータのリスト

cs_name

既定値 : なし

値の範囲 : 文字列

=====

SSL/TLS 通信を実行する際に設定する OpenSSL の Cipher Suites オプションを指定します (TLS 1.3 で使用されます)。指定するオプションの名前は下記 URL に規定される名前を使用します。

<https://www.openssl.org/docs/man1.1.1/man1/ciphers.html>
ciphers

"TLS v1.3 cipher suites"で定められた暗号スイート(Cipher Suite)名。

--

例 :

WSSCipherSuites TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256

--

2.1.9 WSSCipherList

=====

対応 OS : Linux.x86 / Windows

書式 : WSSCipherList <clist_value>[<clist_name>]

clist_value

書式 : (NONE | <openssl_cipher_list>)

既定値 : NONE

値の範囲 : OpenSSL で規定される Cipher List パラメータ

clist_name

既定値 : なし

値の範囲 : 文字列

=====

SSL/TLS 通信を実行する際に設定する OpenSSL の Cipher List オプションを指定します (TLS 1.2 もしくはそれ以前で使用されます)。指定するオプションの名前は下記 URL に規定される名前を使用します。

<https://www.openssl.org/docs/man1.1.1/man1/ciphers.html>
ciphers

"CIPHER LIST FORMAT"及び"CIPHER STRINGS"で定められた書式の暗号リスト(Cipher List)。

--

例 :

WSSCipherList RC4-MD5:RC4-SHA:AES128-SHA:AES256-SHA:HIGH:!DSS:!aNULL

--

2.1.10 WSListenAddress

=====

対応 OS : Linux.x86 / Windows

書式 : WSListenAddress <ws_service_addr>[:<ws_service_port>[:<mcd>]][<acl_name>]

ws_service_addr

既定値 : なし

値の範囲 : IP アドレス

ws_service_port

既定値 : 80

値の範囲 : 1 - 65535

mcd

既定値 : 物理コア数 + 1, 論理コア数/2 + 1, 25 or 1

値の範囲 : 1 - 65535

acl_name

既定値 : なし

値の範囲 : 設定ファイル中に記述されたアクセスコントロールリストの名前

=====

クライアントからの接続を受け付ける WebSocket サービス (平文通信) を定義します。

ws_service_addr は、WebSocket サービスの IP アドレスを指定します。

ws_service_port は、同サービスのポート番号を指定します。

mcd は、オプションです。このサービスでクライアントに提供するセッションあたりの接続の数を指定します。1 を指定した場合は、セッションは単一の接続で動作します。2 以上を指定した場合は、接続の受付とセッションのネゴシエーションが完了した後に、指定された数から 1 を差し引いた数だけ追加で接続が行

われ並列で通信を行います。指定しない場合は、CPU の物理コア数、論理コア数及びライセンスから次の何れかの値に決まります。

- 物理コア数 + 1 （物理コア数 < 論理コア数 かつ 物理コア数 > 1 の場合。但し、最大 25）
- 論理コア数/2 + 1 （物理コア数 = 論理コア数 かつ 論理コア数/2 > 1 の場合。但し、最大 25）
- 1 （物理コア数もしくは論理コア数/2 が 1 の場合やライセンスに多重化機能が含まれない場合）

`acl_name` は、オプションです。TCP サービと同様。

```
--  
例 :  
WSListenAddress 0.0.0.0:8080  
--
```

2.1.11 ListenServiceBonding

=====
対応 OS : Linux / Windows

書式 : ListenServiceBonding <service_name>[... <service_name>]

`service_name`

既定値 : なし

値の範囲 : 設定した `TCPListenAddress` もしくは `UDPListenAddress` の名前
=====

セッションの多重接続機能において複数のサービスを結束（ボンディング）する設定を行います。結束したいサービス名を一つ以上指定します。サービス名は、`-t` オプションで確認します。

セッションで TCP 及び HpFP をハイブリッドで使用する場合などに利用します。

本オプションは、設定ファイル中に一つのみ記述可能です。

クライアントからの接続は通常通りサーバのホスト名とポート番号（TCP もしくは HpFP のいずれかのポート番号）を指定して行います。セッション確立後は、指定されたサービスで利用可能な接続数の合計から 1 を差し引いた数だけ追加で接続が行われます。

```
--  
例 :  
ListenServiceBonding tcp1 udp1  
--
```


2.1.12 UseServerCertificateSecurity

```
=====
対応 OS : Linux.x86 / Windows
書式 : UseServerCertificateSecurity <flag-available>
```

```
-----
flag-available
既定値 : yes
値の範囲 : yes, no
=====
```

サーバ証明書セキュリティ機能を設定します。**yes** を指定した場合、**ServerCertificateFile** で指定されたサーバ証明書もしくは、**ServerKeyFile** のパスから解決された公開鍵を使用して、クライアントとの間で PKI 技術を基礎とするセキュリティ通信を提供します。

```
--
例 :
UseServerCertificateSecurity no
--
```

2.1.13 RequireServerCertificateSecurity

```
=====
対応 OS : Linux.x86 / Windows
書式 : RequireServerCertificateSecurity <flag-available>
```

```
-----
flag-available
既定値 : yes
値の範囲 : yes, no
=====
```

クライアントに対してサーバ証明書セキュリティ機能によるセキュリティ通信を要求するか設定します。**yes** を指定した場合、クライアントが同機能を使用しない通信（平文通信）を要求した場合、接続を拒否します。

```
--
例 :
RequireServerCertificateSecurity no
--
```

2.1.14 ServerKeyFile

```
=====
対応 OS : Linux.x86 / Windows
書式 : ServerKeyFile <file-path>[ <serv-key-name>]
```

```
-----
file-path
```

既定値 :

/etc/hcp/key/server.key (Linux.x86)

C:/ProgramData/Clealink/HCP Tools/key/server.key (Windows)

値の範囲 : ファイルシステムのパス文字列

serv-key-name

既定値 : なし

値の範囲 : 文字列
=====

file-path は、サーバ証明書セキュリティ機能で使用するサーバの秘密鍵のパスを指定します。秘密鍵のパスに接尾辞".pub"を付加したパスに公開鍵が存在する場合は、ServerCertificateFile の設定を使用せずにこの鍵ペアを使用してセキュリティ通信を提供します。

serv-key-name は、オプションです。このサーバの秘密鍵の設定に名前を指定します。WebSocket の暗号通信サービス(WSSListenAddress)から参照するために使用します。

--

例 :

ServerKeyFile /etc/hcp/key/server.key

ServerKeyFile /etc/hcp/key/server.wss.key wss-key

--

クライアントではサーバに初めてアクセスすると、下記の様に公開鍵をknown_hosts に登録するか確認されます。

A secure connection for host 127.0.0.1 can't be established.

RSA key fingerprint is SHA256: 0fzb9DY4qxXWPm/L/4cBKkk+FQ9577NIRYxRquZ6eW
A=.

Are you sure you want to continue connecting [yes/no] ?

登録を許可すると、次のパスに確認済みのホストとして記録されます。次回以降、同じホストの同じ公開鍵であれば確認を求められなくなります。

<ユーザホームディレクトリ>/_hcp/known_hosts (Linux)

<ユーザホームディレクトリ>/_hcp/known_hosts (Windows)

2.1.15 ServerCertificateFile

=====

対応 OS : Linux.x86 / Windows

書式 : ServerCertificateFile <file-path>[<serv-cert-name>]

file-path

既定値 :

/etc/hcp/cert/server.crt (Linux.x86)

C:/ProgramData/Clealink/HCP Tools/cert/server.crt (Windows)

値の範囲 : ファイルシステムのパス文字列

serv-cert-name

既定値 : なし

値の範囲 : 文字列

=====

file-path は、サーバ証明書のパスを指定します。

serv-cert-name は、オプションです。このサーバの証明書の設定に名前を指定します。**WebSocket** の暗号通信サービス(**WSSListenAddress**)から参照するために使用します。

WebSocket の暗号通信サービスを使用する場合は、中間証明書が含まれるとその中間証明書も使用されます。中間証明書を含む場合は、サーバ証明書、中間証明書の順に格納してください。

--

例 :

ServerCertificateFile /etc/hcp/cert/server.crt

ServerCertificateFile /etc/hcp/cert/server.wss.crt **wss-cert**

--

2.1.16 **ServerCertificateChainFile**

=====

対応 OS : Linux.x86 / Windows

書式 : **ServerCertificateChainFile** <file-path>

file-path

既定値 :

/etc/hcp/cert/chain.crt (Linux.x86)

C:/ProgramData/Clealink/HCP Tools/cert/chain.crt (Windows)

値の範囲 : ファイルシステムのパス文字列

=====

サーバ証明書の中間証明書を複数指定します。指定されたファイルに踏まれる中間証明書はクライアントに送信されるサーバ証明書に続けてファイルに保管されている順に中間証明書として送信されます。

--

例 :

```
ServerCertificateChainFile /etc/hcp/cert/chain.crt
```

```
--
```

2.1.17 LocalPasswordAuthentication

```
=====
対応 OS : Linux.x86 / Windows
```

```
書式 : LocalPasswordAuthentication <flag-available>
```

```
-----
flag-available
```

```
既定値 : no
```

```
値の範囲 : yes, no
=====
```

LPA 認証を設定します。yes を指定すると LPA 認証を有効にします。本認証方式を含め、一つでも認証を有効にした場合、クライアントからアクセスするには認証が必要になります（匿名アクセス禁止）。

```
--
```

```
例 :
```

```
LocalPasswordAuthentication yes
```

```
--
```

2.1.18 PAMAuthentication

```
=====
対応 OS : Linux.x86
```

```
書式 : PAMAuthentication <flag-available>
```

```
-----
flag-available
```

```
既定値 : yes
```

```
値の範囲 : yes, no
=====
```

PAM 認証を設定します。yes を指定すると PAM 認証を有効にします。

```
--
```

```
例 :
```

```
PAMAuthentication no
```

```
--
```

PAM 認証は、Linux プラットホーム向けの RPM パッケージで配布されるソフトウェアで有効です。

PAM 認証は、システムの構成に依存します。サービスが動作するオペレーティングシステムの環境にあわせて、次の様な PAM 設定ファイルを適切に構成します。

```
/etc/pam.d/hcpd
```

2.1.19 PubkeyAuthentication

```
=====
対応 OS : Linux.x86 / Windows
書式 : PubkeyAuthentication <flag-available>
```

```
-----
flag-available
既定値 : yes (Linux.x86), no (Windows)
値の範囲 : yes, no
=====
```

RSA 認証を設定します。yes を指定すると RSA 認証を有効にします。

```
--
例 :
PubkeyAuthentication no
--
```

2.1.20 WinLogonUserAuthentication

```
=====
対応 OS : Windows
書式 : WinLogonUserAuthentication <flag-available>
```

```
-----
flag-available
既定値 : yes
値の範囲 : yes, no
=====
```

Windows 認証を設定します。yes を指定すると Windows 認証を有効にします。

```
--
例 :
WinLogonUserAuthentication no
--
```

Windows 認証は、Windows プラットホーム向けに配布されるソフトウェアで有効です。

Windows 認証は、システム標準の API の認証機能へ入力されたユーザ ID（ドメイン名を含む）とパスワードをそのまま引き渡して行われます。

2.1.21 PerformSystemAuthenticationRegardlessUsers

```
=====
対応 OS : Linux.x86 / Windows
書式 : PerformSystemAuthenticationRegardlessUsers <flag-available>
```

```
-----
flag-available
```

既定値 : no

値の範囲 : yes, no

=====

システムの認証(PAM 認証、Windows 認証)をユーザ定義ファイル(/etc/hcp/users)の指定に関わらず実行するか設定します。

--

例 :

PerformSystemAuthenticationRegardlessUsers yes

--

2.1.22 UserDirectoryFallbackAvailable

=====

対応 OS : Linux.x86 / Windows

書式 : UserDirectoryFallbackAvailable <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no

=====

ユーザのホームディレクトリ、作業ディレクトリなどの後退 (Fallback) を許可するか設定します。認証の結果により認識されたホームディレクトリが存在しない場合などに、ルートドキュメントに記載されたディレクトリに後退する動作をするか制御します。本オプションは、クライアントが 1.1.0 以降のソフトウェアの場合は、常に無効(no)として動作します。

--

例 :

UserDirectoryFallbackAvailable yes

--

2.1.23 RejectOnUserHomeDirectoryNotFound

=====

対応 OS : Linux.x86 / Windows

書式 : RejectOnUserHomeDirectoryNotFound <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no

=====

ユーザのホームディレクトリが見つからない場合にアクセスを拒否するか設定します。本オプションは、クライアントが 1.2.0 より以前のソフトウェアの場合は、常に有効(yes)として動作します。

```
--
例 :
RejectOnUserHomeDirectoryNotFound yes
--
```

2.1.24 UsePrivilegeSeparation

```
=====
対応 OS : Linux.x86 / Windows
書式 : UsePrivilegeSeparation <flag-available>
```

```
-----
flag-available
既定値 : yes
値の範囲 : yes, no
=====
```

クライアントセッションに対する特権分離を設定します。yes を指定した場合、同セッションでの処理をサーバ待機プロセスとは別のプロセスで実行します。分離されたプロセス上では認証結果に基づくユーザの資格情報(UID/GID)が実効権限として設定されます。

```
--
例 :
UsePrivilegeSeparation no
--
```

特権分離を無効にした場合は、クライアントセッション上の処理はサービスの実行権限に従って動作します。

Windows サービスでは、Windows ログオン認証以外で認証された場合（LPA 認証、もしくは RSA 認証）は、サービスの実行権限で動作します（特権分離は適用されません）。

2.1.25 PrivilegeSeparationMinimumUID

```
=====
対応 OS : Linux.x86 / Windows
書式 : PrivilegeSeparationMinimumUID <min-uid>
```

```
-----
min-uid
既定値 : 0
値の範囲 : 符号なし整数値
=====
```

特権分離時にセッションを実行できる UID の最小値を設定します。特別な権限を所有するユーザによる実行などを抑制するために使用可能です。

次の項目のいずれかが設定された場合は、本設定は無効となります。

- AllowUsers
- AllowGroups
- DenyUsers
- DenyGroups

--

例 :

PrivilegeSeparationMinimumUID 1000

--

2.1.26 PrivilegeSeparationMinimumGID

=====

対応 OS : Linux.x86 / Windows

書式 : PrivilegeSeparationMinimumGID <min-gid>

min-gid

既定値 : 0

値の範囲 : 符号なし整数値

=====

特権分離時にセッションを実行できる GID の最小値を設定します。特別な権限を所有するユーザによる実行などを抑制するために使用可能です。

次の項目のいずれかが設定された場合は、本設定は無効となります。

- AllowUsers
- AllowGroups
- DenyUsers
- DenyGroups

--

例 :

PrivilegeSeparationMinimumGID 1000

--

2.1.27 PrivilegeSeparationUser

=====

対応 OS : Linux.x86 / Windows

書式 : PrivilegeSeparationUser <username>

username

既定値 : なし

値の範囲 : システムに存在するユーザ名

=====

特権分離を有効にした場合に、適用するユーザ権限が特定されなかった場合に適用するユーザの資格情報を設定します。

指定しない場合は、プラットフォームに依存したユーザ名 (**nobody** など) に解決されます。

--

例 :

PrivilegeSeparationUser nobody

--

2.1.28 PrivilegeSeparationUmask

特権分離を有効にした場合に、ユーザ認証が行われて分離されたプロセスに適用する **Umask** の値を設定します。

書式 **<umask_val>[<dir_umask_val>]**

* **dir_umask_val** は実験的パラメータです。

umask_val には適用する **umask** 値を指定します。**dir_umask_val** を指定しない場合は、ファイル及びディレクトリともにプロセスレベルで適用されます。

既定値 **0022**

値の範囲 **0000** から **0777** の 8 進数

dir_umask_val にはディレクトリに適用したい **umask** 値を指定します。この設定を使用した場合は、**umask_val** はファイルに適用され、**dir_umask_val** はディレクトリに適用されます。プロセスには共通する **umask** 値(**umask_val & dir_umask_val**)が適用され、ファイル及びディレクトリ作成時にアプリケーション(HCP)により共通の値からの差分がそれぞれ適用されます。アプリケーションがハンドルしないファイルやディレクトリ作成では期待通りの **umask** 値が適用されませんのでご注意ください (このパラメータは実験的機能です)。

既定値 なし

値の範囲 **0000** から **0777** の 8 進数

ユーザ毎に **umask** 値を設定したい場合は、**/etc/hcp/users** に設定を記述します。

--

例 :

PrivilegeSeparationUmask 0002

--

2.1.29 PrivilegeSeparationUmaskAnonymous

特権分離を有効にした場合に、ユーザ認証が行われずに（匿名アクセスで）分離されたプロセスに適用する Umask の値を設定します。

書式 <umask_val>[<dir_umask_val>]

* dir_umask_val は実験的パラメータです。

umask_val には適用する umask 値を指定します。dir_umask_val を指定しない場合は、ファイル及びディレクトリともにプロセスレベルで適用されます。

既定値 0002

値の範囲 0000 から 0777 の 8 進数

dir_umask_val にはディレクトリに適用したい umask 値を指定します。この設定を使用した場合は、umask_val はファイルに適用され、dir_umask_val はディレクトリに適用されます。プロセスには共通する umask 値(umask_val & dir_umask_val)が適用され、ファイル及びディレクトリ作成時にアプリケーション(HCP)により共通の値からの差分がそれぞれ適用されます。アプリケーションがハンドルしないファイルやディレクトリ作成では期待通りの umask 値が適用されませんのでご注意ください（このパラメータは実験的機能です）。

既定値 なし

値の範囲 0000 から 0777 の 8 進数

--

例 :

PrivilegeSeparationUmaskAnonymous 0022

--

2.1.30 ApplyUserPermission

=====
対応 OS : Linux.x86 / Windows

書式 : ApplyUserPermission <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no

=====
特権分離を使用しない場合に、認証されたユーザの資格情報(UID/GID)を転送先のファイルパーミッションに適用するか設定します。

```
--  
例 :  
ApplyUserPermission yes  
--
```

2.1.31 AllowUsers

```
=====
対応 OS : Linux.x86
書式 : AllowUsers [<username-pattern>...]
-----
username-pattern
書式 : <name-pattern>[@<host-pattern>]
-----
name-pattern
既定値 : なし
値の範囲 : ワイルドカード ("*"及び"?") を含む文字列
-----
host-pattern
既定値 : なし
値の範囲 : CIDR 形式のネットワークアドレス文字列
=====
```

ログインを許可するシステムユーザ名のパターンを指定します（複数可）。

username-pattern には、ユーザ名のパターンとオプションとして CIDR 形式のネットワークアドレスを指定します。

name-pattern には、ユーザ名を表すパターン文字列を指定します。このパターンには、任意の文字列を表すワイルドカード "*" と任意の一文字を表すワイルドカード "?" を使用することができます。接続したユーザが入力したユーザ名がこのパターンに一致するとログインが許可され認証が実行されます。

host-pattern には、CIDR 形式のネットワークアドレス文字列を指定します。このネットワークアドレス文字列が指定された場合は、ユーザ名が一致し、かつアクセス元ホストが指定されたネットワークアドレスに含まれる場合にログインが許可されます。

このパターンの評価で一致することが確認されると、その時点で全体の評価が終了しログインは許可されます。**DenyUsers** が指定された場合は、**DenyUsers**、**AllowUsers** の順に評価されます。**AllowUsers** または **AllowGroups** が指定された場合、いずれの評価でもパターンの一致が確認されなかった場合は、ログインは拒否されます。

この項目が設定されると、次の設定項目は無効となります。

- PrivilegeSeparationMinimumUID
- PrivilegeSeparationMinimumGID

--

例 :

AllowUsers seg1-*@192.168.0.0/24

--

2.1.32 AllowGroups

=====
対応 OS : Linux.x86

書式 : AllowGroups [<groupname-pattern>...]

groupname-pattern

既定値 : なし

値の範囲 : ワイルドカード ("*"及び"?") を含む文字列

=====

ログインを許可するシステムグループ名のパターンを指定します（複数可）。

groupname-pattern には、グループ名を表すパターン文字列を指定します。任意の文字列を表すワイルドカード "*" と任意の一文字を表すワイルドカード "?" を使用することができます。所属するプライマリグループ名もしくは補助グループ名が指定したパターンに一致するとログインが許可され認証が実行されます。

このパターン評価で一致することが確認されると、その時点で全体の評価が終了しログインは拒否されます。**DenyGroups** が指定された場合は、**DenyGroups**, **AllowGroups** の順に評価されます。**AllowUsers** または **AllowGroups** が指定された場合、いずれの評価でもパターンの一致が確認されなかった場合は、ログインは拒否されます。

この項目が設定されると、次の設定項目は無効となります。

- PrivilegeSeparationMinimumUID
- PrivilegeSeparationMinimumGID

--

例 :

AllowGroups seg1-users

--

2.1.33 DenyUsers

=====
対応 OS : Linux.x86

書式 : DenyUsers [<username-pattern>...]

username-pattern

書式 : <name-pattern>[@<host-pattern>]

name-pattern

既定値 : なし

値の範囲 : ワイルドカード ("*"及び"?") を含む文字列

host-pattern

既定値 : なし

値の範囲 : CIDR 形式のネットワークアドレス文字列

ログインを許可しないシステムユーザ名のパターンを指定します（複数可）。

username-pattern には、ユーザ名のパターンとオプションとして CIDR 形式のネットワークアドレスを指定します。

name-pattern には、ユーザ名を表すパターン文字列を指定します。このパターンには、任意の文字列を表すワイルドカード "*" と任意の一文字を表すワイルドカード "?" を使用することができます。接続したユーザが入力したユーザ名がこのパターンに一致するとログインが拒否され認証が失敗します。

host-pattern には、CIDR 形式のネットワークアドレス文字列を指定します。このネットワークアドレス文字列が指定された場合は、ユーザ名が一致し、かつアクセス元ホストが指定されたネットワークアドレスに含まれる場合にログインが拒否されます。

AllowUsers が指定された場合は、**DenyUsers**, **AllowUsers** の順に評価されます。また、パターンの評価で一致することが確認されると、その時点で全体の評価が終了しログインは拒否されます。

この項目が設定されると、次の設定項目は無効となります。

- **PrivilegeSeparationMinimumUID**
- **PrivilegeSeparationMinimumGID**

--
例 :

DenyUsers seg1-*@192.168.0.0/24

2.1.34 DenyGroups

対応 OS : Linux.x86

書式 : **DenyGroups** [<groupname-pattern>...]

----- groupname-pattern

既定値 : なし

値の範囲 : ワイルドカード ("*"及び"?") を含む文字列
=====

ログインを許可しないシステムグループ名のパターンを指定します (複数可)。

groupname-pattern には、グループ名を表すパターン文字列を指定します。任意の文字列を表すワイルドカード "*" と任意の一文字を表すワイルドカード "?" を使用することができます。所属するプライマリグループ名もしくは補助グループ名が指定したパターンに一致するとログインが拒否され認証が失敗します。

AllowGroups が指定された場合は、**DenyGroups**, **AllowGroups** の順に評価されます。また、パターンの評価で一致することが確認されると、その時点で全体の評価が終了しログインは拒否されます。

この項目が設定されると、次の設定項目は無効となります。

- **PrivilegeSeparationMinimumUID**
- **PrivilegeSeparationMinimumGID**

--

例 :

DenyGroups guest-users

--

2.1.35 AuthorizedKeySearchDir

=====

対応 OS : Linux.x86 / Windows

書式 : **AuthorizedKeySearchDir** <search-dir>

search-dir

既定値 :

/etc/hcp/authkeys (Linux.x86)

C:/ProgramData/Clealink/HCP Tools/authkeys (Windows)

値の範囲 : ファイルシステムのパス文字列
=====

RSA 認証の際にユーザの公開鍵を特定するために探索するディレクトリを指定します。

--

例 :

AuthorizedKeySearchDir /etc/hcp/authkeys

--

指定されたディレクトリで次の様なファイル名を公開鍵が保管されたファイルと見做して探索します。

<ユーザ名>.pub

2.1.36 AuthorizedKeyFile

=====
対応 OS : Linux.x86 / Windows

書式 : AuthorizedKeyFile <file-path>

file-path

既定値 :

~/hcp/authorized_keys (Linux.x86)

~/_hcp/authorized_keys (Windows)

値の範囲 : ユーザディレクトリを表すパス表記(~、チルダ)を含むファイルパス
=====

RSA 認証の際にユーザの公開鍵を特定するために探索するユーザホームディレクトリ内の鍵保管ファイルの書式を指定します。

2.1.37 CACertificateFile

=====
対応 OS : Linux.x86 / Windows

書式 : CACertificateFile <file-path>

file-path

既定値 :

/etc/hcp/cacert.pem (Linux.x86)

C:/ProgramData/Clealink/HCP Tools/cacert.pem (Windows)

値の範囲 : ファイルシステムのパス文字列
=====

クライアント認証に使用する CA 証明書及び中間証明書が保管されているファイルを指定します。

--

例 :

CACertificateFile /etc/hcp/cacert.pem

--

PEM 形式の証明書をサポートします。

2.1.38 CACertificatePath (予約)

=====
対応 OS : Linux.x86 / Windows

書式 : CACertificatePath <dir-path>

dir-path

既定値 :

/etc/ssl (Linux.x86)

C:/ProgramData/Clealink/HCP Tools/ssl (Windows)

値の範囲 : ファイルシステムのパス文字列

クライアント認証に使用する CA 証明書及び中間証明書が保管されているディレクトリを指定します。

--

例 :

CACertificatePath /etc/ssl

--

PEM 形式の証明書をサポートします。

2.1.39 CARevocationFile

対応 OS : Linux.x86 / Windows

書式 : CARevocationFile <file-path>

file-path

既定値 :

/etc/hcp/crl.pem (Linux.x86)

C:/ProgramData/Clealink/HCP Tools/crl.pem (Windows)

値の範囲 : ファイルシステムのパス文字列

クライアント認証に使用する CRL が保管されているファイルを指定します。

--

例 :

CARevocationFile /etc/hcp/crl.pem

--

PEM 形式の失効リスト(CRL)をサポートします。

2.1.40 CARevocationPath (予約)

対応 OS : Linux.x86 / Windows

書式 : CARevocationPath <dir-path>

dir-path

既定値 :

/etc/ssl (Linux.x86)

C:/ProgramData/Clealink/HCP Tools/ssl (Windows)

値の範囲 : ファイルシステムのパス文字列

=====

クライアント認証に使用する CRL が保管されているディレクトリを指定します。

--

例 :

CARevocationPath /etc/ssl

--

PEM 形式の失効リスト(CRL)をサポートします。

2.1.41 OCSPRevocationEnabled

=====

対応 OS : Linux.x86 / Windows

書式 : OCSPRevocationEnabled <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no

=====

クライアント認証で使用する OCSP (オンライン証明書失効検査プロトコル) を設定します。yes を指定すると有効にします。

--

例 :

OCSPRevocationEnabled no

--

2.1.42 LocalUserFile

=====

対応 OS : Linux.x86 / Windows

書式 : LocalUserFile <file-path>[<usage>]

file-path

既定値 :

/etc/hcp/users (Linux.x86)

C:/ProgramData/Clealink/HCP Tools/users (Windows)

値の範囲 : ファイルシステムのパス文字列

usage

既定値 : overwrite

値の範囲 : **overwrite, define**

ユーザ情報を定義したファイルと利用方法を指定します。

file-path には定義ファイルのパスを指定します。

usage にはこの定義ファイルの使用方法を次のいずれかで指定します。

- **overwrite**

- **define**

overwrite を指定した場合は、ユーザの認証を行う際に既存のシステムユーザの認証方法を上書きする情報として認識します。

define を指定した場合は、本ソフトウェアで扱うと許可されたユーザ情報として認識します。ファイルに記載されないユーザについては認証を拒否します。

--

例 :

LocalUserFile /etc/hcp/users define

--

HCP tools の認証では、このファイルに定義されたユーザを認証において認識します（定義されないユーザで認証を試行した場合は失敗します）。

2.1.43 LocalPasswordFile

対応 OS : **Linux.x86 / Windows**

書式 : **LocalPasswordFile <file-path>**

file-path

既定値 :

/etc/hcp/passwd (Linux.x86)

C:/ProgramData/Clealink/HCP Tools/passwd (Windows)

値の範囲 : ファイルシステムのパス文字列

LPA 認証で使用するユーザの資格情報（パスワードハッシュ）を定義したファイルを指定します。

--

例 :

LocalPasswordFile /etc/hcp/passwd

--

2.1.44 AcceptableCryptMethod

対応 OS : Linux.x86 / Windows

書式 : AcceptableCryptMethod <method-names>

method-names

書式 : <method-name>[...]

既定値 : AES256/CTR/VMAC AES256/CBC AES128/CBC

method-name

値の範囲 : PLAIN, AES128/CBC, AES192/CBC, AES256/CBC, AES128/CBC/HMAC,
AES192/CBC/HMAC, AES256/CBC/HMAC, AES128/CBC/VMAC, AES192/CBC/VMAC,
AES256/CBC/VMAC, AES128/CBC/VMAC64, AES192/CBC/VMAC64, AES256/CBC/VMAC64,
AES128/CTR/HMAC, AES192/CTR/HMAC, AES256/CTR/HMAC, AES128/CTR/VMAC,
AES192/CTR/VMAC, AES256/CTR/VMAC, AES128/CTR/VMAC64, AES192/CTR/VMAC64,
AES256/CTR/VMAC64, AES128/GCM (予約), AES192/GCM (予約), AES256/GCM
(予約)

=====

暗号アルゴリズムを設定します。

AES128/CBC と指定した場合は、AES128/CBC/HMAC と解釈されます（これらは同一のアルゴリズムです。AES192/CBC 及び AES256/CBC についても同様です）。

新たに追加したアルゴリズム（CTR モード、VMAC モードを含むアルゴリズム）に対応していないバージョン（1.2.5 以前）と通信する場合は、これらの新しいアルゴリズムは接続時のネゴシエーションで無視されます（エラーとはなりません）。

--

例 :

AcceptableCryptMethod AES256/CBC PLAIN

--

クライアントとの間で通信するメッセージの暗号化に使用されます。どのアルゴリズムが使用されるかは、クライアントの設定と併せて評価し、一致したアルゴリズムを使用します。

2.1.45 AcceptableDigestMethod

=====

対応 OS : Linux.x86 / Windows

書式 : AcceptableDigestMethod <method-names>

method-names

書式 : <method-name>[...]

既定値 : MM128 SHA256 SHA160

method-name

値の範囲 : NONE, SHA160, SHA224, SHA256, SHA384, SHA512, MD5, MM32, MM128
=====

ダイジェストアルゴリズムを設定します。

--

例 :

AcceptableDigestMethod SHA256 MM128 NONE

--

クライアントとの間で通信するメッセージ、ファイルやそのデータブロックの検証に使用されます。どのアルゴリズムが使用されるかは、クライアントの設定と併せて評価し、一致したアルゴリズムを使用します。

また、データ検査に HMAC を使用する暗号通信 (AES256/CBC/HMAC など) を行う場合、セキュリティダイジェストではないアルゴリズム (MD5, MM32, MM128) は、指定されなかったものとして扱われます。

2.1.46 RequireDataIntegrityChecking

=====

対応 OS : Linux.x86 / Windows

書式 : RequireDataIntegrityChecking <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no
=====

クライアントとの間で暗号化通信を行う場合に、サーバが MAC(Message Authentication Code)による通信メッセージの検査 (データ完全性検査) を要求するか設定します。

no を指定した場合、クライアントが通信メッセージの検査を行わないことを要求した場合に、その要求を受け入れて通信を継続します。

yes を指定した場合、この要求を拒否します。

通常は yes (デフォルト) で使用してください。前述の通信メッセージの検査が省略されることを理解したうえでご利用ください。通常、暗号通信のパフォーマンス向上の目的で使用します。

```
--
例 :
RequireDataIntegrityChecking no
--
```

2.1.47 TransportCharEncoding

```
=====
対応 OS : Linux.x86 / Windows
書式 : TransportCharEncoding <encodings>
-----
```

encodings

書式 : <encoding>[...]

既定値 : UTF8

encoding

値の範囲 : US-ASCII, UTF8, UTF16, UTF32
=====

トランスポートで使用する文字列のエンコーディング方式を指定します。

```
--
例 :
TransportCharEncoding UTF8 UTF16 US-ASCII
--
```

クライアントとの間で交換するファイルパスなどの文字列に適用されます。どのエンコーディングが使用されるかは、クライアントの設定と併せて評価し、一致したエンコーディングを使用します。

2.1.48 HostEncoding

```
=====
対応 OS : Linux.x86 / Windows
書式 : HostEncoding <encoding>
-----
```

encoding

既定値 :

UTF-8 (Linux)

CP932 (Windows)

値の範囲 : システムかつエンコーディング変換ライブラリでサポートされるエンコーディング名 (プラットフォーム依存)
=====

ホストで使用されている文字列エンコーディングを指定します。

```
--
例 :
```

HostEncoding EUC-JP

--

ファイルパスなどをソフトウェアの内部文字列表現へ変換する際に使用されます。

2.1.49 HeaderCompress (予約)

2.1.50 ContentCompress (予約)

2.1.51 MaxConcurrentThread

```
=====
対応 OS : Linux.x86
書式 : MaxConcurrentThread <max-threads>
-----
max-threads
既定値 : 0
値の範囲 : 符号付整数
=====
```

スレッド数制限を設定します。

2.1.52 MaxTotalConnection

```
=====
対応 OS : Linux.x86 / Windows
書式 : MaxTotalConnection <max-total-con>
-----
max-total-con
既定値 : 150
値の範囲 : 符号付整数
=====
```

接続数制限を設定します。

--

例 :

MaxTotalConnection 5

--

2.1.53 MaxTcpConnection

```
=====
対応 OS : Linux.x86 / Windows
書式 : MaxTcpConnection <max-tcp-con>
-----
max-tcp-con
既定値 : 50
```

値の範囲 : 符号付整数

TCP 接続数制限を設定します。

--

例 :

MaxTcpConnection 5

--

2.1.54 MaxUdpConnection

HpFP(UDP)接続数制限を設定します。

対応 OS : Linux.x86 / Windows

書式 : MaxUdpConnection <max-udp-con>

max-udp-con

既定値 : 50

値の範囲 : 符号付き整数

--

例 :

MaxUdpConnection 5

--

2.1.55 MaxWsConnection

WebSocket 接続数制限を設定します。

対応 OS : Linux.x86 / Windows

書式 : MaxWsConnection <max-ws-con>

max-ws-con

既定値 : 50

値の範囲 : 符号付き整数

--

例 :

MaxWsConnection 5

--

2.1.56 MaxConnectionPerUser

=====
対応 OS : Linux.x86 / Windows

書式 : MaxConnectionPerUser <max-con-per-user>

max-con-per-user

既定値 : 50

値の範囲 : 符号付き整数
=====

ユーザ毎の接続数制限を設定します。

--

例 :

MaxConnectionPerUser 1

--

2.1.57 MaxConnectionPerSec

=====
対応 OS : Linux.x86 / Windows

書式 : MaxConnectionPerSec <max-con-per-sec>

max-con-per-sec

既定値 : 25

値の範囲 : 符号付き整数
=====

秒間の接続数制限を設定します。

--

例 :

MaxConnectionPerSec 10

--

2.1.58 MaxReceiveFileSize

=====
対応 OS : Linux.x86 / Windows

書式 : MaxReceiveFileSize <file-size>

file-size

既定値 : 8EB - 1B (符号付倍長整数最大値。制限なし)

値の範囲 : 符号付き倍長整数
=====

受信を許可する最大のファイルサイズを設定します。

--

例 :

MaxReceiveFileSize 1GB

--

2.1.59 MaxSendFileSize

=====

対応 OS : Linux.x86 / Windows

書式 : MaxSendFileSize <file-size>

file-size

既定値 : 8EB - 1B (符号付倍長整数最大値。制限なし)

値の範囲 : 符号付き倍長整数

=====

送信を許可する最大のファイルサイズを設定します。

--

例 :

MaxSendFileSize 1GB

--

2.1.60 MaxRequestFileEntryAtOnce

=====

対応 OS : Linux.x86 / Windows

書式 : MaxRequestFileEntryAtOnce <max-file-req-at-once>

max-file-req-at-once

既定値 : 50

値の範囲 : 符号付き整数

=====

ファイル要求の一括送信を許可する最大の数を設定します。

--

例 :

MaxRequestFileEntryAtOnce 1000

--

2.1.61 MaxTotalBufferSize

=====

対応 OS : Linux.x86 / Windows

書式 : MaxTotalBufferSize <max-total-buf-size>

max-total-buf-size

既定値 : 4GB

値の範囲 : 符号付き倍長整数

=====
ファイルのデータ処理に使用できる最大のメモリバッファサイズを設定します。

--

例 :

MaxTotalBufferSize 8GB

--

2.1.62 MaxBufferSizePerConnection

=====
対応 OS : Linux.x86 / Windows

書式 : MaxBufferSizePerConnection <max-buf-size-per-con>

max-buf-size-per-con

既定値 : 100MB

値の範囲 : 符号なし倍長整数

=====
クライアントセッション毎のファイルのデータ処理に使用できる最大のメモリバッファサイズを設定します。

--

例 :

MaxBufferSizePerConnection 512MB

--

2.1.63 MaxTotalReceiveRate

=====
対応 OS : Linux.x86 / Windows

書式 : MaxTotalReceiveRate <bandwidth>

bandwidth

既定値 : 10Gbit

値の範囲 : 符号なし倍長整数

=====
トランスポートの受信帯域のシェーピング (制限) を設定します。

--

例 :

MaxTotalReceiveRate 1Gbit

--

本機能は、TCP/HpFP(UDP)層とアプリケーション層の間で帯域制限を行います。指定値が 5Gbps を超える場合は、無制限（シェーピングなし）に設定されます（他の帯域シェーピングオプションも同様）。

2.1.64 MaxTotalSendRate

```
=====
対応 OS : Linux.x86 / Windows
書式 : MaxTotalSendRate <bandwidth>
```

```
-----
bandwidth
既定値 : 10Gbit
値の範囲 : 符号なし倍長整数
=====
```

トランスポートの送信帯域のシェーピング（制限）を設定します。

```
--
例 :
MaxTotalSendRate 1Gbit
--
```

2.1.65 MaxReceiveRatePerConnection

```
=====
対応 OS : Linux.x86 / Windows
書式 : MaxReceiveRatePerConnection <bandwidth>
```

```
-----
bandwidth
既定値 : 10Gbit
値の範囲 : 符号なし倍長整数
=====
```

クライアントセッション毎のトランスポート受信帯域のシェーピング（制限）を設定します。

```
--
例 :
MaxReceiveRatePerConnection 100Mbit
--
```

2.1.66 MaxSendRatePerConnection

```
=====
対応 OS : Linux.x86 / Windows
書式 : MaxSendRatePerConnection <bandwidth>
```

```
-----
bandwidth
```

既定値 : 10Gbit

値の範囲 : 符号なし倍長整数

=====

クライアントセッション毎のトランスポート送信帯域のシェーピング（制限）を設定します。

--

例 :

MaxSendRatePerConnection 100Mbit

--

2.1.67 InitHeaderBlockSize

=====

対応 OS : Linux.x86 / Windows

書式 : InitHeaderBlockSize <block-size>

block-size

既定値 : 50KB

値の範囲 : 符号なし倍長整数

=====

初期ヘッダブロックのサイズを設定します。

--

例 :

InitHeaderBlockSize 10KB

--

ファイル要求などのメッセージを複数含むヘッダブロックが形成可能な上限のサイズを与えます。このオプションは通信開始直後に適用されます。

2.1.68 InitContentBlockSize

=====

対応 OS : Linux.x86 / Windows

書式 : InitContentBlockSize <block-size>

block-size

既定値 : 1MB

値の範囲 : 符号なし倍長整数

=====

初期コンテンツブロックのサイズを設定します。

--

例 :

InitContentSize 2MB

--

ファイルのデータを複数含むコンテンツブロックが形成可能な上限のサイズを与えます。このオプションは通信開始直後に適用されます。

2.1.69 MaxHeaderBlockSize

=====
対応 OS : Linux.x86 / Windows

書式 : MaxHeaderBlockSize <block-size>

block-size

既定値 : 50KB

値の範囲 : 符号なし倍長整数
=====

ヘッダブロックのサイズを拡張する最大のサイズを設定します。

--

例 :

MaxHeaderBlockSize 100KB

--

通信を開始すると消費帯域を評価して形成可能なヘッダブロックのサイズを増減させます。本オプションは、このサイズを増加させることができる上限値を与えます。

2.1.70 MaxContentSize

=====
対応 OS : Linux.x86 / Windows

書式 : MaxContentSize <block-size>

block-size

既定値 : 1MB

値の範囲 : 符号なし倍長整数
=====

コンテンツブロックのサイズを拡張する最大のサイズを設定します。

10Gbps を超える環境などで通信性能が頭打ちになった場合などに、InitContentSize と併せて変更すると性能が改善する場合があります。

--

例 :

MaxContentSize 4MB

--

通信を開始すると消費帯域を評価して形成可能なコンテンツブロックのサイズを増減させます。本オプションは、このサイズを増加させることができる上限値を与えます。

2.1.71 FileLock

```
=====
対応 OS : Linux.x86 / Windows
書式 : FileLock <flag-available>
```

```
-----
flag-available
既定値 : no
値の範囲 : yes, no
=====
```

ファイルの読み書きにファイルロックを使用するか設定します。

yes を設定した場合、ファイルロックを獲得してからファイルの読み書きを行います。**Linux** プラットフォームではアドバイザリロックを使用するため、本ソフトウェア間もしくは同じロック機構を使用するソフトウェアの間で有効に働きます。

no に設定した場合は、ファイルのロックを獲得せずにファイルデータの読み書きを実行します。**NFS** などのネットワークファイルシステムでハングアップが発生する場合やロック獲得時にエラーが発生する場合は、**no** に設定します。**no** で使用する場合は、hcp コマンドで指定する宛先の重複にご注意ください。

ロック要求の処理は適切に動作していて、単にロックの獲得を待機している場合は、次の様なログが記録されるか（FileLockTrials が 0） 、

```
2018/07/05 16:34:10 00007f638bd97740:INFO :Acquiring a lock to the file w
as rejected at the first trial.
2018/07/05 16:34:13 00007f638bd97740:INFO :Acquiring a lock to the file c
ontinues to be rejected about few seconds.
```

試行回数の上限に達したことを検出して処理を中断します（FileLockTrials が 0 以上）。

```
--
例 :
FileLock yes
--
```

2.1.72 FileLockTrials

```
=====
対応 OS : Linux.x86 / Windows
書式 : FileLockTrials <num-trials>
```

num-trials

既定値 : 0

値の範囲 : 符号なし整数

ファイルロックの獲得を試行する回数を設定します。0 を指定するとロックを獲得するまで待機（ブロック）します。

--

例 :

FileLockTrials 5

--

旧名 FileLockRetries も使用可能です。

2.1.73 FileLockTrialInterval

対応 OS : Linux.x86 / Windows

書式 : FileLockTrialInterval <trial-interval>

trial-interval

既定値 : 3

値の範囲 : 符号なし整数

ファイルロックの要求間隔を設定します（秒単位）。

--

例 :

FileLockTrialInterval 10

--

旧名 FileLockRetryInterval も使用可能です。

2.1.74 AtomicLikeSaving

対応 OS : Linux.x86 / Windows

書式 : AtomicLikeSaving <flag-available> <temp-file-suffix>[<temp-file-prefix>]

flag-available

既定値 : no

値の範囲 : yes, no

temp-file-suffix

既定値 : .tmp
値の範囲 : 16 文字までの文字列, NONE, RANDn

temp-file-prefix

既定値 : NONE
値の範囲 : 16 文字までの文字列, NONE, RANDn

=====

ファイル転送時のファイル書き込みで行われるアトミックライクな保存（一時ファイルを経由する二段階保存）の設定を行います。

flag-available は、この設定を有効にするか指定します。

temp-file-suffix は、生成される一時ファイルの接尾辞（サフィックス）を指定します。“NONE”を指定した場合は、接尾辞を付加しません。“RANDn”を指定した場合は、n 文字の乱数生成された文字列を付加します。文字数を指定しない場合は 6 文字の文字列を生成します。

temp-file-prefix は、生成される一時ファイルの接頭辞（プレフィックス）を指定します。“NONE”を指定した場合は、接頭辞を付加しません。“RANDn”を指定した場合は、n 文字の乱数生成された文字列を付加します。文字数を指定しない場合は 6 文字の文字列を生成します。

Linux 版では一時ファイルを生成する際に、最終的な書き込み先に書き込みを行う権限があるか検査されます。

また、再開機能(-r オプション)を使用する場合、途中まで転送していたファイルは先頭から再度転送しなおします。

接頭辞及び接尾辞の文字列を乱数生成する場合は、既存のファイルと名前が重複しない文字列が使用されます。

2.1.75 AtomicLikeSavingThreshold

=====

対応 OS : Linux.x86 / Windows
書式 : AtomicLikeSavingThreshold <threshold>

threshold

既定値 : 100KB
値の範囲 : 符号付倍長整数

=====

アトミックライクな保存を適用するファイルの閾値をサイズで指定します。

指定したサイズ未満のファイルにはこの保存方法は適用されません。0 バイトを指定した場合は、全てのファイルに適用されます。

2.1.76 AtomicLikeSavingRejectOverwriteRequest

=====
対応 OS : Linux.x86 / Windows

書式 : AtomicLikeSavingRejectOverwriteRequest <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no
=====

アトミックライクな保存において生成される一時ファイルを上書きする要求（クライアントのコマンドで指定します）を拒否するかどうか設定します。

アプリケーションのファイル名の規則と接尾辞や接頭辞の構成により一時ファイルとの衝突の恐れがない場合などに、no に変更しクライアントからの上書き要求を許可するように設定します。

2.1.77 TransportTimeout

=====
対応 OS : Linux.x86 / Windows

書式 : TransportTimeout <timeout>

timeout

既定値 : 180

値の範囲 : 符号なし整数
=====

トランスポートのタイムアウトを秒単位で設定します。

--

例 :

TransportTimeout 60

--

2.1.78 IdleTimeout

=====
対応 OS : Linux.x86 / Windows

書式 : IdleTimeout <timeout>

timeout

既定値 : 600

値の範囲 : 符号なし整数

セッションのアイドルタイムアウトを秒単位で設定します。接続上でオペレーション（コマンド実行を単位とする処理）が指定時間実行されない場合に、サーバから接続を切断します。クライアントがターミナル機能（複数のコマンドを任意の時点で実行可能。API ライブラリで提供されます）を使用している場合に適用されます。

```
--
例 :
IdleTimeout 180
--
```

2.1.79 DocPoint

対応 OS : Linux.x86 / Windows
書式 : DocPoint <doc_point_name>

doc_point_name
既定値 : なし
値の範囲 : 文字列

ドキュメントポイントを定義します。doc_point_name は、このドキュメントポイントの名称を記述します。

最初に定義されたドキュメントポイントは、ユーザのホームディレクトリが解決できなかった場合の既定のホームディレクトリとして使用されます。

```
--
例 :
DocPoint /home
--
```

2.1.80 DocPath

対応 OS : Linux.x86 / Windows
書式 : DocPath <doc_path>

doc_path
既定値 : なし
値の範囲 : ファイルシステムのパス文字列

ドキュメントポイントが指すディレクトリパスを指定します。

--

例 :

DocPath /home

--

この項目で指定されたディレクトリパスのファイルやディレクトリへのアクセスを許可します。

2.1.81 PermitAccessRead

=====

対応 OS : Linux.x86 / Windows

書式 : PermitAccessRead <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no

=====

ドキュメントポイントでの読み込みアクセスを設定します。

--

例 :

PermitAccessRead yes

--

no を指定した場合、このドキュメントポイント内のファイルの読み出しは禁止されます。ファイル転送を行う場合は、転送元のファイルの読み出しなどが読み込みエラーとなります。

2.1.82 PermitAccessWrite

=====

対応 OS : Linux.x86 / Windows

書式 : PermitAccessWrite <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no

=====

ドキュメントポイントでの書き込みアクセスを設定します。

--

例 :

```
PermitAccessWrite yes
```

```
--
```

no を指定した場合、このドキュメントポイント内のファイルへの書出しは禁止されます。ファイル転送を行う場合は、転送先のファイルへの書出しなどが書込みエラーとなります。

2.1.83 PermitAccessOverwrite

```
=====  
対応 OS : Linux.x86 / Windows
```

```
書式 : PermitAccessOverwrite <flag-available>  
-----
```

```
flag-available
```

```
既定値 : no
```

```
値の範囲 : yes, no  
=====
```

ドキュメントポイントでの上書きアクセスを設定します。

```
--
```

```
例 :
```

```
PermitAccessOverwrite yes
```

```
--
```

no を指定した場合、このドキュメントポイント内のファイルへの上書きは禁止されます。ファイル転送を行う場合は、転送先のファイルへ上書き（既に存在するファイルへのデータ書込み）などが書込みエラーとなります。

2.1.84 PermitAccessDelete

```
=====  
対応 OS : Linux.x86 / Windows
```

```
書式 : PermitAccessDelete <flag-available>  
-----
```

```
flag-available
```

```
既定値 : no
```

```
値の範囲 : yes, no  
=====
```

ドキュメントポイントでの削除アクセスを設定します。

```
--
```

```
例 :
```

```
PermitAccessDelete yes
```

```
--
```

no を指定した場合、このドキュメントポイント内のファイルの削除は禁止されます。

2.1.85 PermitAccessRandomRead (予約)

=====
対応 OS : Linux.x86 / Windows

書式 : PermitAccessRandomRead <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no
=====

ドキュメントポイントでのランダム読み込みアクセスを設定します。

--

例 :

PermitAccessRandomRead yes

--

2.1.86 PermitAccessRandomWrite (予約)

=====
対応 OS : Linux.x86 / Windows

書式 : PermitAccessRandomWrite <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no
=====

ドキュメントポイントでのランダム書き込みアクセスを設定します。

--

例 :

PermitAccessRandomRead yes

--

2.1.87 AccessList

=====
対応 OS : Linux.x86 / Windows

書式 : AccessList <acl_name>

acl_name

既定値 : なし

値の範囲 : 文字列
=====

アクセスコントロールリストを定義します。 **acl_name** はオプションです。アクセスコントロールリストの名前を指定します。

```
--
例：
AccessList acl1
--
```

acl-name を指定しない場合は、無名のアクセスコントロールリストとして扱われます。この無名のアクセスコントロールリストは、1 つだけ定義できます。

2.1.88 Allow

```
=====
対応 OS : Linux.x86 / Windows
書式 : Allow (<ip_addr> <net_mask>|any)[ <hpfp_cong_mode_modifier>]
-----
ip_addr
既定値 : なし
値の範囲 : IP アドレス
-----
net_mask
既定値 : なし
値の範囲 : サブネットマスク表記
-----
hpfp_cong_mode_modifier
書式 : <modifier>[...]
modifier := (+|-)(M|S|A)[...]
既定値 : なし
=====
```

アクセスコントロールリスト内に、許可アクセスを定義します。

ip_addr は、IP アドレスを指定します。

net_mask は、ネットマスクを指定します。

キーワード **any** は、全てのネットワークを表します。

hpfp_cong_mode_modifier は、HpFP の輻輳制御モードの上書きを指定します。

+は、記述したルールで接続を受け付けた場合に、直後に記載した輻輳制御モードの許可を追加します。

-は、直後に記載した輻輳制御モードの許可を取り消します。

M、S および A は、それぞれ HpFP の輻輳制御モードの MODEST、FAIR_FAST_START および AGGRESSIVE を表します。

```
--
例：
    Allow 192.168.1.0 255.255.255.0 -A+M
--
```

接続元の IP アドレスが指定されたネットワークに含まれる場合、そのアクセスを許可します。

2.1.89 Deny

```
=====
対応 OS : Linux.x86 / Windows
書式 : Deny (<ip_addr> <net_mask>|any)
-----
ip_addr
既定値 : なし
値の範囲 : IP アドレス
-----
net_mask
既定値 : なし
値の範囲 : サブネットマスク表記
=====
```

アクセスコントロールリスト内に、拒否アクセスを定義します。

```
--
例：
    Deny 192.168.1.0 255.255.255.0
--
```

接続元の IP アドレスが指定されたネットワークに含まれる場合、そのアクセスを拒否します。

2.1.90 SyslogOption

```
=====
対応 OS : Linux.x86 / Windows
書式 : SyslogOption <syslog-options>
-----
syslog-options
書式 : syslog-option[ ...]
既定値 : CONS PID
-----
syslog-option
```

値の範囲 : CONS, NDELAY, NOWAIT, ODELAY, PERROR, PID

syslog のオプションを設定します（複数指定可）。

各オプションは、接頭辞”LOG_“を付加した syslog のオプションに対応します。

```
--
例 :
SyslogOption PID
--
```

2.1.91 SyslogFacility

対応 OS : Linux.x86 / Windows

書式 : SyslogFacility <syslog-facility>

syslog-facility

既定値 : DAEMON

値の範囲 : AUTH, CRON, DAEMON, FTP, LOCAL0 - LOCAL7, LPR, MAIL, NEWS, USER, UUCP

syslog のファシリティを設定します。

各ファシリティは、接頭辞”LOG_“を付加した syslog のファシリティに対応します。

```
--
例 :
SyslogFacility FTP
--
```

2.1.92 SystemLog

対応 OS : Linux.x86 / Windows

書式 : SystemLog <log-level>[<flag-available>][<log-rotation-conf>][<log-path>]

log-level

既定値 : INFO

値の範囲 : EMERG, ALERT, CRIT, ERR, WARNING, INFO, DEBUG

flag-available

既定値 : yes

値の範囲 : yes, no

log-rotation-conf

書式 : (**FileSize** <file-size> <backups> | **DatePattern** <date-pattern>)

file-size

既定値 : なし

値の範囲 : 符号あり倍長整数

backups

既定値 : なし

値の範囲 : 符号なし整数

date-pattern

既定値 : なし

値の範囲 : yyyy-MM, yyyy-MM-dd, yyyy-MM-dd-HH, yyyy-MM-dd-HH-mm

log-path

既定値 : /var/log/hcpd.log

値の範囲 : ファイルシステムのパス文字列

システムのログ設定を行います。

log-level は、ログのレベルを指定します。

flag-available に **yes** を指定するとログを出力します。

log-rotation-conf は、ログのローテーションを指定します。指定しない場合は、ローテーションしません。**FileSize** を指定するとファイルサイズを閾値としたローテーションを行い、**DatePattern** を指定すると日時にもとづくローテーションを行います。

file-size は、ローテーションを行いファイルサイズの閾値をバイト単位で指定します。

backups は、**FileSize** を指定してローテーションを行う場合にローテーションされたファイルを保持する世代数の上限を指定します。

date-pattern は、日時でローテーションするパターンを指定します。

FileSize を指定した場合は、ログのパス名に次の様に世代数を接尾辞として付加してファイルをローテーションします。

<指定されたログのパス> // 現在書込み中のログのパス

<指定されたログのパス>.1

<指定されたログのパス>.2

...

<指定されたログのパス>.n // backups に n を指定

backups で指定された世代数を超えるファイルは、ローテーション実行時に削除されます。

DatePattern を指定した場合は、ログのパス名に次の様に指定されたパターンに従って日時を接尾辞として付加してファイルをローテーションします。

// yyyy-MM-dd の場合

<指定されたログのパス> // 現在書込み中のログのパス

<指定されたログのパス>.2019-12-10 // 2019/12/10 のログ

<指定されたログのパス>.2019-12-09

...

<指定されたログのパス>.2019-11-30

...

ローテーションは、指定されたパターンで定まる日時の区間を単位に行います。月単位の場合は、その月の 1 日 0 時 0 分 0 秒から翌月の同時刻が到来する直前までになります。

例：

2019/11/01 00:00:00 から 2019/12/01 00:00:00 まで

(但し、2019/12/01 00:00:00 を含まない)

分単位の場合は、その分の 0 秒から次の分の 0 の直前までになります。

例：

2019/11/01 10:30:00 から 2019/11/01 10:31:00 まで

(但し、2019/11/01 10:31:00 を含まない)

ローテーションは、各区間を経過後にログの書込みが要求されるとその書込みの前に実行されます。ファイル名は、該当する区間の接尾辞が付加された名前に変更されます。

// yyyy-MM-dd-HH-mm の場合のローテーションの事例

2019/12/10 00:00 サーバ起動

2019/12/10 00:05 サーバ停止

2019/12/10 00:07 サーバ再起動

...

--

<指定されたログのパス>

<指定されたログのパス>.2019-12-10-00-11

<指定されたログのパス>.2019-12-10-00-10 // 00:09 から 00:10 の間は書込みがなかった

```
<指定されたログのパス>.2019-12-10-00-08
<指定されたログのパス>.2019-12-10-00-07 // 再起動後の 00:07 のログ
<指定されたログのパス>.2019-12-10-00-05 // サーバ停止までの 00:05 のログ (再起動時にファイルの更新日時から判断してローテーション実施)
...
<指定されたログのパス>.2019-12-10-00-00
```

但し、特権分離を使用しているサーバでは定期的に（128 ミリ秒程度の間隔で）ローテーションが実行されます。ローテーションが遅延して実行される場合があるため、通常は次の区間に出力されるログが現在の区間に出力されることがあります。

log-path は、ログのパスを指定します。コマンドラインパラメータの **-l** とともに指定した場合は、コマンドラインパラメータの指定が適用されます。

```
--
例 1 :
SystemLog WARNING FileSize 10MB 10
例 2 :
SystemLog WARNING DatePattern yyyy-MM-dd
例 3 :
SystemLog WARNING // SystemLogLevel に同じ
--
```

2.1.93 SystemLogLevel

```
=====
対応 OS : Linux.x86 / Windows
書式 : SystemLogLevel <log-level>
```

```
-----
log-level
既定値 : INFO
値の範囲 : EMERG, ALERT, CRIT, ERR, WARNING, INFO, DEBUG
=====
```

システムのログレベルを設定します。**syslog** の機能には影響しません。

SystemLog と共に記述した場合は、ログレベルの値のみ上書きされます。

```
--
例 :
SystemLogLevel WARNING
--
```

2.1.94 ApplicationStatLog

=====
 対応 OS : Linux.x86 / Windows

書式 : ApplicationStatLog <flag-available>[<log-rotation-conf>]

flag-available

既定値 : yes

値の範囲 : yes, no

log-rotation-conf

書式 : (FileSize <file-size> <backups> | DatePattern <date-pattern>)

file-size

既定値 : なし

値の範囲 : 符号あり倍長整数

backups

既定値 : なし

値の範囲 : 符号なし整数

date-pattern

既定値 : なし

値の範囲 : yyyy-MM, yyyy-MM-dd, yyyy-MM-dd-HH, yyyy-MM-dd-HH-mm
 =====

アプリケーション統計を設定します。

flag-available に **yes** を指定するとアプリケーションの統計情報を出力します。

log-rotation-conf は、出力のローテーションを指定します。**SystemLog** で設定するローテーションと同様の動作を行います。但し、特権分離を使用している場合に実行される定期的なローテーションは行われません。

FileSize と **DatePattern** でそれぞれ統計ログの基準パスとして指定されたパスをもとに次の様にローテーションが行われます。

// **FileSize** の場合

<指定されたパス>.application

<指定されたパス>.application.1

<指定されたパス>.application.2

...

<指定されたパス>.application.n

// **DatePattern** の場合

<指定されたパス>.application

```
<指定されたパス>.application.2019-12-10
<指定されたパス>.application.2019-12-09
...
```

ローテーションされたファイルに統計情報のヘッダは含まれません。

```
--
例 1 :
ApplicationStatLog no
例 2 :
ApplicationStatLog yes FileSize 10MB 10
例 3 :
ApplicationStatLog yes DatePattern yyyy-MM-dd
--
```

2.1.95 TransportStatLog

=====
対応 OS : Linux.x86 / Windows

書式 : TransportStatLog <flag-available>[<log-rotation-conf>]

flag-available

既定値 : no

値の範囲 : yes, no

log-rotation-conf

書式 : (FileSize <file-size> <backups> | DatePattern <date-pattern>)

file-size

既定値 : なし

値の範囲 : 符号あり倍長整数

backups

既定値 : なし

値の範囲 : 符号なし整数

date-pattern

既定値 : なし

値の範囲 : yyyy-MM, yyyy-MM-dd, yyyy-MM-dd-HH, yyyy-MM-dd-HH-mm
=====

トランスポート統計を設定します。

flag-available に yes を指定するとトランスポートの統計情報を出力します。

log-rotation-conf は、出力のローテーションを指定します。SystemLog で設定するローテーションと同様の動作を行います。SystemLog と同様に特権分離を使用している場合は、定期的なローテーションが実行されます。

FileSize と DatePattern でそれぞれ統計ログの基準パスとして指定されたパスをもとに次の様にローテーションが行われます。

```
// FileSize の場合
<指定されたパス>.transport.tcp.service_<サービス番号>.<サービスポート番号>.th
read_<スレッド番号>
<指定されたパス>.transport.tcp.service_<サービス番号>.<サービスポート番号>.th
read_<スレッド番号>.1
<指定されたパス>.transport.tcp.service_<サービス番号>.<サービスポート番号>.th
read_<スレッド番号>.2
...
<指定されたパス>.transport.tcp.service_<サービス番号>.<サービスポート番号>.th
read_<スレッド番号>.n
// DatePattern の場合
<指定されたパス>.transport.tcp.service_<サービス番号>.<サービスポート番号>.th
read_<スレッド番号>
<指定されたパス>.transport.tcp.service_<サービス番号>.<サービスポート番号>.th
read_<スレッド番号>.2019-12-10
<指定されたパス>.transport.tcp.service_<サービス番号>.<サービスポート番号>.th
read_<スレッド番号>.2019-12-09
...
```

ローテーションされたファイルに統計情報のヘッダは含まれません。

```
--
例 1 :
TransportStatLog yes
例 2 :
TransportStatLog yes FileSize 10MB 10
例 3 :
TransportStatLog yes DatePattern yyyy-MM-dd
--
```

2.1.96 SystemStatLog

=====
対応 OS : Linux.x86 / Windows

書式 : SystemStatLog <flag-available>[<log-rotation-conf>]

flag-available

既定値 : yes

値の範囲 : yes, no

log-rotation-conf

書式 : (FileSize <file-size> <backups> | DatePattern <date-pattern>)

file-size

既定値 : なし

値の範囲 : 符号あり倍長整数

backups

既定値 : なし

値の範囲 : 符号なし整数

date-pattern

既定値 : なし

値の範囲 : yyyy-MM, yyyy-MM-dd, yyyy-MM-dd-HH, yyyy-MM-dd-HH-mm
=====

システム統計を設定します。

flag-available に yes を指定するとシステムの統計情報を出力します。

log-rotation-conf は、出力のローテーションを指定します。SystemLog で設定するローテーションと同様の動作を行います。但し、特権分離を使用している場合に実行される定期的なローテーションは行われません。

FileSize と DatePattern でそれぞれ統計ログの基準パスとして指定されたパスをもとに次の様にローテーションが行われます。

// FileSize の場合

<指定されたパス>.system

<指定されたパス>.system.1

<指定されたパス>.system.2

...

<指定されたパス>.system.n

// DatePattern の場合

<指定されたパス>.system

<指定されたパス>.system.2019-12-10

<指定されたパス>.system.2019-12-09

...

ローテーションされたファイルに統計情報のヘッダは含まれません。

--

例 1 :

SystemStatLog yes

例 2 :

SystemStatLog yes FileSize 10MB 10

例 3 :

SystemStatLog yes DatePattern yyyy-MM-dd

--

2.1.97 FileOperationLog

=====

対応 OS : Linux.x86 / Windows

書式 : FileOperationLog <flag-available>[<log-rotation-conf>][<log-path>]

flag-available

既定値 : no

値の範囲 : yes, no

log-rotation-conf

書式 : (FileSize <file-size> <backups> | DatePattern <date-pattern>)

file-size

既定値 : なし

値の範囲 : 符号あり倍長整数

backups

既定値 : なし

値の範囲 : 符号なし整数

date-pattern

既定値 : なし

値の範囲 : yyyy-MM, yyyy-MM-dd, yyyy-MM-dd-HH, yyyy-MM-dd-HH-mm

log-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

=====

ファイル操作をロギングする機能を設定します。

flag-available に yes を指定するとファイル操作のログを出力します。

log-rotation-conf は、出力のローテーションを指定します。SystemLog で設定するローテーションと同様の動作を行います。

FileSize と **DatePattern** でそれぞれ統計ログの基準パスとして指定されたパスをもとに次の様にローテーションが行われます。

```
// FileSize の場合
<ログのパス>
<ログのパス>.1
<ログのパス>.2
...
<ログのパス>.n
// DatePattern の場合
<ログのパス>
<ログのパス>.2019-12-10
<ログのパス>.2019-12-09
...
```

ローテーションされたファイルに統計情報のヘッダは含まれません。

log-path は、ログを出力するファイルのパスを指定します。指定しない場合は、次の値が使用されます。

```
/var/log/hcpd.file.operation.log (Linux.x86)
C:/ProgramData/Clealink/HCP Tools/hcpd.file.operation.log (Windows)
```

ファイルの操作ログでは、次のファイル I/O 処理を記録します。

- ファイル読み込み終了
- ファイル書き込み終了
- ファイル削除（同期による削除を含む）
- ディレクトリ作成
- ファイルリネーム
- ハードリンク作成
- シンボリックリンク作成
- ファイルリスト参照

また、アプリケーションの送達確認を含めた記録として次の様な情報も記録されます。

- ファイル転送アップロード完了
- ファイル転送ダウンロード完了
- ファイル同期完了
- ファイル削除完了
- ディレクトリ作成完了

- ファイルリネーム完了
- ハードリンク作成完了
- シンボリックリンク作成完了

共通する情報として、次の項目が記録されます。

- 日時
- アクセス元 IP 及びポート番号
- ユーザ名

ファイルパスは、各処理毎に固有の内容が記録されます。

ログは次の書式で出力されます。

```
=====
書式:
yyyy/mm/dd HH:MM:SS.<usec> <remote-ip> <username> <hcp-operation-name>[ <
sub-operation-label>] <path>...
yyyy/mm/dd HH:MM:SS.<usec> <remote-ip> <username> <file-io-operation-name
>\[<hcp-operation-name>\] <path>...
-----
usec
値の範囲 : マイクロ秒(000000 - 999999)
-----
remote-ip
値の範囲 : アクセス元 IP 及びポート番号
-----
username
値の範囲 : ユーザ名 (/etc/hcp/users に定義のユーザ名) 及び認証方式
-----
hcp-operation-name
値の範囲 : FT, FS, FR, LR, DC, FM, FL
-----
sub-operation-label
値の範囲 : U, D, H, S
-----
file-io-operation-name
値の範囲 : FileRead, FileWritten, FileDeleted, DirectoryCreated,
FileRenamed, LinkCreated, SymbolicLinkCreated, ListFilesRawFormat
=====
```

hcp-operation-name は、次のアプリケーションの種別を区別するラベルを表します。

- FT (ファイル転送)

- FS (ファイル同期)
- FR (ファイル削除)
- LR (ファイル一覧 ls, dir 出力)
- DC (ディレクトリ作成)
- FM (ファイル移動)
- FL (リンク作成)

sub-operation-label は、特定のアプリケーションの種別で処理の種類を分類するためのラベルを表します。

- U (アップロード。FT で表示)
- D (ダウンロード。FT で表示)
- H (ハードリンク作成。FL で表示)
- S (シンボリックリンク作成。FL で表示)

file-io-operation-name は、ファイルの I/O 操作の種別を区別するラベルを表します。

- FileRead (ファイル読み込み終了)
- FileWritten (ファイル書き込み終了)
- FileDeleted (ファイル削除実施済み)
- DirectoryCreated (ディレクトリ作成済み)
- FileRenamed (ファイル名変更済み)
- LinkCreated (ハードリンク作成済み)
- SymbolicLinkCreated (シンボリックリンク作成済み)
- ListFilesRawFormat (ls もしくは dir 実行予定)

--

出力例:

```
2020/01/31 10:34:52.277120 127.0.0.1:51660 user[PAM] FileWritten[FT] /home/user/file_nodiskio_0
2020/01/31 10:34:52.277175 127.0.0.1:51660 user[PAM] FT U /home/user/file_nodiskio_0
2020/01/31 10:35:14.946750 127.0.0.1:51662 user[PAM] FileRead[FT] /home/user/file_nodiskio_0
2020/01/31 10:35:15.002770 127.0.0.1:51662 user[PAM] FT D /home/user/file_nodiskio_0
2020/01/31 10:35:30.002700 127.0.0.1:51662 user[PAM] FS /home/user/dir_sync/stat.log
2020/01/31 10:35:30.013558 127.0.0.1:51664 user[PAM] FileDeleted[FS] /home/user/dir_sync/stat.log
2020/01/31 10:35:47.713558 127.0.0.1:51664 user[PAM] FileDeleted[FR] /home
```

```
e/user/stat.3.log
2020/01/31 10:35:47.765413 127.0.0.1:51664 user[PAM] FR /home/user/stat.3.
log
2020/01/31 10:38:45.686206 127.0.0.1:51670 user[PAM] DirectoryCreated[DC]
/home/user/hmkdir13
2020/01/31 10:38:45.789370 127.0.0.1:51670 user[PAM] DC /home/user/hmkdir
13
2020/01/31 10:39:22.411968 127.0.0.1:51674 user[PAM] FileRenamed[FM] /hom
e/user/stat.log /home/user/stat2.log
2020/01/31 10:39:22.463710 127.0.0.1:51674 user[PAM] FM /home/user/stat.l
og /home/user/stat2.log
2020/01/31 10:40:00.087660 127.0.0.1:51678 user[PAM] SymbolicLinkCreated
[FL] /home/user/stat2.log /home/user/stat.log
2020/01/31 10:40:00.165831 127.0.0.1:51678 user[PAM] FL S /home/user/stat
2.log /home/user/stat.log
2020/01/31 10:40:13.693415 127.0.0.1:51680 user[PAM] LinkCreated[FL] /hom
e/user/stat2.log /home/user/stat.h.log
2020/01/31 10:40:13.746160 127.0.0.1:51680 user[PAM] FL H /home/user/stat
2.log /home/user/stat.h.log
2020/02/06 13:54:21.282066 127.0.0.1:50186 user[PAM] ListFilesRawFormat[L
R] /home/user/hmkdir4
2020/02/06 13:54:21.282104 127.0.0.1:50186 user[PAM] ListFilesRawFormat[L
R] /home/user/hmkdir5
--
--
例 1 :
FileOperationLog yes
例 2 :
FileOperationLog yes FileSize 10MB 10
例 3 :
FileOperationLog yes DatePattern yyyy-MM-dd
例 4 :
FileOperationLog yes FileSize 10MB 10 /var/tmp/hcpd.file.operation.log
--
```

2.1.98 CallbackScript

```
=====
対応 OS : Linux.x86 / Windows
書式 : CallbackScript <flag-available>[ <script-path>[ <data-store-path
>]]
-----
flag-available
既定値 : no
値の範囲 : yes, no
-----
```

script-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

data-store-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

アプリケーションの実行が終了したタイミングで呼び出すスクリプト（プログラム）を設定します。

flag-available に **yes** を指定するとスクリプトの呼出を有効にします。

script-path は、スクリプトのパスを指定します。チルダ(~)を先頭に使用すると、アプリケーションを実行しているユーザ（認証されたユーザ）のホームディレクトリを展開して使用します。

data-store-path は、スクリプトの実行時に与えられるデータ（パラメータ情報や実行結果）を保存するパスを指定します。チルダ(~)を先頭に使用すると、アプリケーションを実行しているユーザ（認証されたユーザ）のホームディレクトリを展開して使用します。

全てのアプリケーションの処理が終了すると、その直後に次の様な書式でコマンドを実行します。

=====
コマンド書式 : <script-path> <exit-code> <start-date-and-time> <end-date-and-time> <remote-ip> <username> <hcp-operation-name> <param-saved-path> <output-saved-path>

script-path

値の範囲 : ファイルシステムのパス文字列（チルダ展開済み）

exit-code

値の範囲 : 終了コード

start-date-and-time

値の範囲 : 次の書式の日時 YYYY/MM/DD hh:mm:ss

end-date-and-time

値の範囲 : 次の書式の日時 YYYY/MM/DD hh:mm:ss

remote-ip

値の範囲 : アクセス元 IP アドレス及びポート番号

username

値の範囲 : ユーザ名 (/etc/hcp/users に定義のユーザ名) 及び認証方式

hcp-operation-name

値の範囲 : hcp, hrm, hcp-ls, hmkdir, hpwd, hmv, hln, transfer, remove, listraw, mkd, pwd, move, link, cwd

param-saved-path

値の範囲 : ファイルシステムのパス文字列

output-saved-path

値の範囲 : ファイルシステムのパス文字列

script-path には、hcpd.conf で設定されたスクリプトパスにチルダ(~)展開が実施されたパスが使用されます。

exit-code には、実行したアプリケーションの終了結果を表す理由コードが渡されます。アプリケーション統計に記録される理由コードと同じものが渡されます。

start-date-and-time 及び **end-date-and-time** には、それぞれアプリケーションの処理を開始した日時と終了した日時が渡されます。

hcp-operation-name には、次のアプリケーション (API オペレーション) の種別を区別するラベルが渡されます。

- hcp (ファイル転送コマンド)
- hrm (ファイル削除コマンド)
- hcp-ls (ファイル一覧コマンド)
- hmkdir (ディレクトリ作成コマンド)
- hpwd (ワーキングディレクトリ出力コマンド)
- hmv (ファイル移動コマンド)
- hln (リンク作成コマンド)
- transfer (API ファイル転送処理)
- remove (API ファイル削除処理)
- listraw (API ファイル一覧処理)
- mkd (API ディレクトリ作成処理)
- pwd (API ワーキングディレクトリ取得処理)
- move (API ファイル移動処理)
- link (API リンク作成処理)

param-saved-path には、入力パラメータの情報が保存されたファイルのパスが渡されます。このファイルには、クライアントで確認できる実行結果記録(.hcp.out)のうちサーバ上で特定された指定されたパス情報と処理のオプションが記録されます。

--

出力例:

```
[user@localhost ~]$ cat .hcp/callback/hcp.cb.20200206_152252_468.13521.param
OPT copy_mode ALLCOPY
OPT overwrite_mode FORCE
OPT fail_action_mode HALT
OPT preserve_permission no
OPT recursive yes
OPT any_dirs no
OPT regex no
OPT verify_payload no
OPT copy_symlink no
OPT follow_symlink no
OPT no_copy_empty_file no
OPT no_copy_empty_dir no
OPT no_copy_dot_file no
OPT no_copy_dot_dir no
OPT copy_hidden no
OPT check_archive no
OPT resuming no
OPT no_app_io yes num_files 1 file_size 1024
OPT no_sess_io no
SRC /home/user
--
```

output-saved-path には、実行結果の情報が保存されたファイルのパスが渡されます。このファイルには、クライアントで確認できる実行結果記録(.hcp.out)のうち各ファイルの実行結果に相当する内容が記録されます。

--

出力例:

```
[user@localhost ~]$ cat .hcp/callback/hcp.cb.20200206_152252_468.13521.out
OK 0000 FT 00000001 /home/user/file_nodiskio_0
--
```

--

例:

```
CallbackScript yes /var/tmp/hcp_callback.sh /var/tmp/hcp_callback
```

--

2.1.99 MemoryTransferConcurrency

=====
 対応 OS : Linux.x86 / Windows

書式 : MemoryTransferConcurrency <num-concur> <wait-type> <busy-sleep-nsec>
 c>

num-concur

既定値 : 自動 (物理コア数 / 2 と 16 のいずれか小さい値)

値の範囲 : 符号なし整数

wait-type

既定値 : cond

値の範囲 : cond, busy

busy-sleep-nsec

既定値 : 1

値の範囲 : 符号なし整数
 =====

ローカル I/O を抑制するモード (hcp コマンド -n オプション。メモリツーメモリ転送動作時) を実行する場合に、送信元で使用する疑似データをデータブロックバッファにコピー (memcpy によるメモリコピー) する際の並列動作を設定します。

num-concur は、並列数を指定します。1 を指定すると並列化せずに標準のメモリコピー処理(memcpy)を実行します。

wait-type は、並列動作するメモリコピースレッドが待機する際に使用する待機方式を指定します。**cond** は条件付き待機(conditional wait)を、**busy** はビジー待機(busy wait)を行います。**busy** を指定した場合は、並列数を指定した数の CPU の使用率が常時 100%まで達する場合があります。

busy-sleep-nsec は、ビジー待機する際に待機する時間をナノ秒で指定します。

このオプションは 100G 環境などでベンチマークを行う際に、シングルスレッドによるメモリコピー(memcpy)の性能限界 (このメモリコピーのボトルネックがアプリ通信性能のボトルネックになる) を解消するための性能チューニングに使用します。

--

例 :

MemoryTransferConcurrency 1 cond 1 # 無効化

MemoryTransferConcurrency 12 busy 1 # ビジー待機、4 並列、1 ナノ秒待機

--

2.1.100 MaxReadRatePerConnection

```
=====
対応 OS : Linux.x86 / Windows
書式 : MaxReadRatePerConnection <read-rate>
```

```
-----
read-rate
既定値 : 10Ebit (無制限)
値の範囲 : 符号なし倍長整数 (10Ebit まで)
=====
```

ファイル転送時にディスクからファイルを読み込む速度の上限を設定します。この設定はファイル転送のセッション毎に適用されます。bps 単位で符号付倍長整数の最大値もしくはそれを超える値が指定された場合は無制限と設定されます。

この設定は SSD などを使用する際に長時間の読み書きを高レートで行うとオーバーヒートを起こすことがあります（特に書込み側）、それを抑制する場合などに使用します。

```
--
例 :
MaxReadRatePerConnection 10Gbit
--
```

2.1.101 MaxWriteRatePerConnection

```
=====
対応 OS : Linux.x86 / Windows
書式 : MaxWriteRatePerConnection <write-rate>
```

```
-----
write-rate
既定値 : 10Ebit (無制限)
値の範囲 : 符号なし倍長整数 (10Ebit まで)
=====
```

ファイル転送時にディスクからファイルを書込む速度の上限を設定します。この設定はファイル転送のセッション毎に適用されます。bps 単位で符号付倍長整数の最大値もしくはそれを超える値が指定された場合は無制限と設定されます。

この設定は SSD などを使用する際に長時間の読み書きを高レートで行うとオーバーヒートを起こすことがあります（特に書込み側）、それを抑制する場合などに使用します。

```
--
例 :
MaxWriteRatePerConnection 10Gbit
--
```

2.1.102 EnsureDestinationInFileTransfer

```
=====
対応 OS : Linux.x86 / Windows
書式 : EnsureDestinationInFileTransfer <flag-available>
-----
flag-available
既定値 : yes
値の範囲 : yes, no
=====
```

ファイル転送で宛先のディレクトリが存在しない場合に、作成を試みるか制御します。本オプションは、1.1.0 以降のソフトウェアでは常に無効(no)として動作します。

```
--
例 :
EnsureDestinationInFileTransfer no
--
```

2.1.103 StatLogPerUserInPrivilegeSeparation

```
=====
対応 OS : Linux.x86 / Windows
書式 : StatLogPerUserInPrivilegeSeparation <flag-available>
-----
flag-available
既定値 : no
値の範囲 : yes, no
=====
```

特権分離を使用している場合の統計ログをユーザ毎に記録するか設定します。

```
--
例 :
StatLogPerUserInPrivilegeSeparation yes
--
```

2.1.104 ApplicationStatLogSecurityEx

```
=====
対応 OS : Linux.x86 / Windows
書式 : ApplicationStatLogSecurityEx <flag-available>
-----
flag-available
既定値 : yes
値の範囲 : yes, no
=====
```

アプリ統計ログでセキュリティに関する詳細情報を出力するか設定します。

--

例 :

ApplicationStatLogSecurityEx no

--

2.2 クライアントコマンド共通設定項目

クライアントコマンドに共通して指定できる設定を記述します。

項目名	説明
ProtocolVersion	プロトコルバージョン(2 固定)
RequireServerCertificateSecurity	サーバ証明書セキュリティを要求
RejectFallbackServerCertificateSecurity	サーバ証明書セキュリティダウングレード禁止
IgnoreCertificateCNInvalid	共通名(Common name)を検査しない
IgnoreCertificateDateInvalid	証明書の期限を検査しない
IgnoreUnknownCA	CA 及び証明書連鎖を検査しない
IgnoreRevocation	証明書の失効を検査しない
WSSIgnoreCertificateCNInvalid	共通名(Common name)を検査しない (WebSocket)
WSSIgnoreCertificateDateInvalid	証明書の期限を検査しない(WebSocket)
WSSIgnoreUnknownCA	CA 及び証明書連鎖を検査しない (WebSocket)
WSSIgnoreRevocation	証明書の失効を検査しない(WebSocket)
WSSOptions	WSS オプション設定
WSSCipherSuites	WSS TLS 1.3 Cipher Suites オプション
WSSCipherList	WSS TLS 1.2 or earlier Cipher Suites オプション
CACertificateFile	CA 証明書ファイル
CACertificatePath	CA 証明書探索ディレクトリ (予約)
CARevocationFile	CRL ファイル
CARevocationPath	CRL 探索ディレクトリ (予約)
OCSPRevocationEnabled	OCSP (オンライン証明書失効検査) 機

	能
StrictHostKeyChecking	サーバホスト鍵受入ポリシー設定
LocalPasswordAuthentication	LPA 認証
PAMAuthentication	PAM 認証
PubkeyAuthentication	RSA 認証
WinLogonUserAuthentication	Windows 認証
PrivateKeySearchDir	秘密鍵探索ディレクトリ
PrivateKeyFile	秘密鍵ファイル
AcceptableCryptMethod	暗号方式
AcceptableDigestMethod	ダイジェスト方式
DisableDataIntegrityChecking	暗号通信メッセージ検査無効設定
AcceptDataIntegrityCheckingOnRejection	暗号通信メッセージ検査無効拒否受入設定
TransportCharEncoding	トランスポート文字エンコーディング
HostEncoding	ホスト文字エンコーディング
CompressLevel	圧縮レベル
HeaderCompress	ヘッダ圧縮
ContentCompress	コンテンツ圧縮
MaxConcurrentThread	使用スレッド数制限(Linux)
MaxReceiveFileSize	受信ファイルサイズ制限
MaxSendFileSize	送信ファイルサイズ制限
MaxRequestFileEntryAtOnce	一括ファイル要求制限
MaxBufferSize	ペイロードバッファメモリ割当制限
MaxReceiveRate	受信レート制限 (接続単位)
MaxSendRate	送信レート制限 (接続単位)
InitHeaderBlockSize	初期ヘッダブロックサイズ
InitContentBlockSize	初期コンテンツブロックサイズ
MaxHeaderBlockSize	最大ヘッダブロックサイズ
MaxContentBlockSize	最大コンテンツブロックサイズ
FileLock	ファイルロック

FileLockTrials	ファイルロック再試行回数
FileLockTrialInterval	ファイルロック試行間隔（秒）
AtomicLikeSaving	アトミック（不可分）ファイル保存
AtomicLikeSavingThreshold	アトミック（不可分）ファイル保存 実行閾値
TransportTimeout	トランスポートタイムアウト
ApplicationLog	ログ設定
ApplicationLogLevel	ログレベル
ApplicationStatLog	アプリケーション統計
TransportStatLog	トランスポート統計
UDPTransportExtensionBufferSize	HpFP トランスポート拡張バッファサイズ設定
TCPTransportSocketSendBuffer	TCP 送信バッファ設定
MemoryTransferConcurrency	メモリ転送並列設定
MaxReadRate	読込レート制限（接続単位）
MaxWriteRate	書込レート制限（接続単位）
PubkeyAuthenticationPrior	RSA 認証優先設定
ApplicationStatLogSecurityEx	アプリ統計ログセキュリティ詳細情報出力設定

2.2.1 RequireServerCertificateSecurity

=====
 対応 OS : Linux / Windows

書式 : RequireServerCertificateSecurity <flag-available>

 flag-available

既定値 : yes

値の範囲 : yes, no
 =====

サーバに対してサーバ証明書セキュリティ機能によるセキュリティ通信を要求するか設定します。yes を指定した場合、サーバが同機能を使用する通信を設定していない場合、接続を拒否します。

--

例 :

RequireServerCertificateSecurity no

--

2.2.2 RejectFallbackServerCertificateSecurity

=====
対応 OS : Linux / Windows

書式 : RejectFallbackServerCertificateSecurity <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no
=====

サーバ証明書セキュリティ機能を使用したセキュリティ通信を確立できない場合に、安全でない通信（平文通信）にフォールバックする動作を禁止します。

--

例 :

RejectFallbackServerCertificateSecurity no

--

2.2.3 IgnoreCertificateCNInvalid

=====
対応 OS : Linux / Windows

書式 : IgnoreCertificateCNInvalid <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no
=====

サーバ証明書の共通名(Common Name)の検査を無効にする設定を行います。yesを指定すると、サーバ証明書の検証を行う際に証明書の共通名が接続に指定したアドレス（もしくはホスト名）と一致するかどうかの確認を行いません。

--

例 :

IgnoreCertificateCNInvalid yes

--

2.2.4 IgnoreCertificateDateInvalid

=====
対応 OS : Linux / Windows

書式 : IgnoreCertificateDateInvalid <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no

=====

証明書の有効期間の検査を無効にする設定を行います。yes を指定すると、証明書の検証を行う際に証明書の有効期間(NotBefore 及び NotAfter)の検査を行いません。

--

例 :

IgnoreCertificateDateInvalid yes

--

2.2.5 IgnoreUnknownCA

=====

対応 OS : Linux / Windows

書式 : IgnoreUnknownCA <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no

=====

CA 証明書の検査を無効にする設定を行います。yes を指定すると、サーバ証明書の中間証明書およびルート証明書に関する検証を行いません。

--

例 :

IgnoreUnknownCA yes

--

2.2.6 IgnoreRevocation

=====

対応 OS : Linux / Windows

書式 : IgnoreRevocation <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no

=====

証明書失効検査 (CRL 検査) を無効にする設定を行います。yes を指定すると、証明書の失効状態の確認を行いません。

--

例 :

IgnoreRevocation yes

--

2.2.7 WSSIgnoreCertificateCNInvalid

=====
対応 OS : Linux / Windows

書式 : WSSIgnoreCertificateCNInvalid <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no
=====

WebSocket 通信時(SSL/TLS)にサーバ証明書の共通名(Common Name)の検査を無効にする設定を行います。yes を指定すると、サーバ証明書の検証を行う際に証明書の共通名が接続に指定したアドレス（もしくはホスト名）と一致するかどうかの確認を行いません。

--

例 :

WSSIgnoreCertificateCNInvalid yes

--

2.2.8 WSSIgnoreCertificateDateInvalid

=====
対応 OS : Linux / Windows

書式 : WSSIgnoreCertificateDateInvalid <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no
=====

WebSocket 通信時(SSL/TLS)に証明書の有効期間の検査を無効にする設定を行います。yes を指定すると、証明書の検証を行う際に証明書の有効期間(NotBefore 及び NotAfter)の検査を行いません。

--

例 :

WSSIgnoreCertificateDateInvalid yes

--

2.2.9 WSSIgnoreUnknownCA

=====
対応 OS : Linux / Windows

書式 : WSSIgnoreUnknownCA <flag-available>

flag-available

既定値 : no

値の範囲 : yes, no
=====

WebSocket 通信時(SSL/TLS)に CA 証明書の検査を無効にする設定を行います。yes を指定すると、サーバ証明書の中間証明書およびルート証明書に関する検証を行いません。

--

例 :

WSSIgnoreUnknownCA yes

--

2.2.10 WSSIgnoreRevocation (予約)

2.2.11 WSSOptions

=====

対応 OS : Linux / Windows

書式 : WSSOptions <opt_value>

opt_value

書式 : (NONE | <openssl_opt_values>)

既定値 : NONE

値の範囲 : OpenSSL で規定される SSL/TLS オプション名のリスト
=====

SSL/TLS 通信を実行する際に設定する OpenSSL オプションを指定します。指定するオプションの名前は下記 URL に規定される名前を使用します。

https://www.openssl.org/docs/man1.1.1/man3/SSL_CTX_set_options.html
SSL_CTX_set_options

--

例 :

WSSOptions SSL_OP_NO_COMPRESSION:SSL_OP_NO_SSLv3

--

2.2.12 WSSCipherSuites

=====

対応 OS : Linux / Windows

書式 : WSSCipherSuites <cs_value>

cs_value

書式 : (NONE | <openssl_cipher_suite_values>)
既定値 : NONE
値の範囲 : OpenSSL で規定される Cipher Suites パラメータのリスト

=====

SSL/TLS 通信を実行する際に設定する OpenSSL の Cipher Suites オプションを指定します (TLS 1.3 で使用されます)。指定するオプションの名前は下記 URL に規定される名前を使用します。

<https://www.openssl.org/docs/man1.1.1/man1/ciphers.html>
ciphers

"TLS v1.3 cipher suites"で定められた暗号スイート(Cipher Suite)名。

--
例 :
WSSCipherSuites TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:TLS_AES_128_GCM_SHA256
--

2.2.13 WSSCipherList

=====

対応 OS : Linux / Windows
書式 : WSSCipherList <clist_value>

clist_value
書式 : (NONE | <openssl_cipher_list>)
既定値 : NONE
値の範囲 : OpenSSL で規定される Cipher List パラメータ

=====

SSL/TLS 通信を実行する際に設定する OpenSSL の Cipher List オプションを指定します (TLS 1.2 もしくはそれ以前で使用されます)。指定するオプションの名前は下記 URL に規定される名前を使用します。

<https://www.openssl.org/docs/man1.1.1/man1/ciphers.html>
ciphers

"CIPHER LIST FORMAT"及び"CIPHER STRINGS"で定められた書式の暗号リスト(Cipher List)。

--
例 :
WSSCipherList RC4-MD5:RC4-SHA:AES128-SHA:AES256-SHA:HIGH:!DSS:!aNULL
--

2.2.14 CACertificateFile

=====
対応 OS : Linux / Windows

書式 : CACertificateFile <file-path>

file-path

既定値 :

/etc/hcp/cacert.pem (Linux)

C:/ProgramData/Clealink/HCP Tools/cacert.pem (Windows)

値の範囲 : ファイルシステムのパス文字列
=====

サーバ証明書の検証に使用する CA 証明書が保管されているファイルのパスを指定します。

--

例 :

CACertificateFile /etc/hcp/cacert.pem

--

PEM 形式の証明書をサポートします。

2.2.15 CACertificatePath (予約)

=====
対応 OS : Linux / Windows

書式 : CACertificatePath <dir-path>

dir-path

既定値 : yes

/etc/ssl (Linux)

C:/ProgramData/Clealink/HCP Tools/ssl (Windows)

値の範囲 : ファイルシステムのパス文字列
=====

サーバ証明書の検証に使用する CA 証明書が保管されているディレクトリのパスを指定します。

--

例 :

CACertificatePath /etc/ssl

--

PEM 形式の証明書をサポートします。

2.2.16 CARevocationFile

=====
対応 OS : Linux / Windows

書式 : CARevocationFile <file-path>

file-path

既定値 : yes

/etc/hcp/crl.pem (Linux)

C:/ProgramData/Clealink/HCP Tools/crl.pem (Windows)

値の範囲 : ファイルシステムのパス文字列
=====

サーバ証明書の検証に使用する失効リスト(CRL)が保管されているファイルを指定します。

--

例 :

CARevocationFile /etc/hcp/crl.pem

--

PEM 形式の失効リスト(CRL)をサポートします。

2.2.17 CARevocationPath (予約)

=====
対応 OS : Linux / Windows

書式 : CARevocationPath <dir-path>

dir-path

既定値 : yes

/etc/ssl (Linux)

C:/ProgramData/Clealink/HCP Tools/ssl (Windows)

値の範囲 : ファイルシステムのパス文字列
=====

サーバ証明書の検証に使用する失効リスト(CRL)が保管されているディレクトリを指定します。

--

例 :

CARevocationPath /etc/ssl

--

PEM 形式の失効リスト(CRL)をサポートします。

2.2.18 WSSCACertificateFile

```
=====
対応 OS : Linux / Windows
書式 : WSSCACertificateFile <file-path>
-----
```

```
file-path
既定値 : なし
値の範囲 : ファイルシステムのパス文字列
=====
```

WebSocket 通信時(SSL/TLS)にサーバ証明書の検証に使用する CA 証明書が保管されているファイルのパスを指定します。省略された場合は、WebSocket 通信ライブラリが既定で検出する CA 証明書が使用されます。

```
--
例 :
WSSCACertificateFile /etc/hcp/cacert.pem
--
```

OpenSSL ツールで出力できる PEM 形式の証明書をサポートします。

2.2.19 OCSPRevocationEnabled

```
=====
対応 OS : Linux / Windows
書式 : OCSPRevocationEnabled <flag-available>
-----
```

```
flag-available
既定値 : yes
値の範囲 : yes, no
=====
```

サーバ証明書の検証に使用する OCSP（オンライン証明書失効検査プロトコル）を設定します。yes を指定すると有効にします。

```
--
例 :
OCSPRevocationEnabled no
--
```

2.2.20 StrictHostKeyChecking

```
=====
対応 OS : Linux / Windows
書式 : StrictHostKeyChecking <switch>
-----
```

```
switch
```

既定値 : ask

値の範囲 : ask, yes, no

=====

サーバホスト鍵の受け入れポリシーを指定します。ask を指定すると、未知の鍵を受信した場合に受け入れるかどうか確認を行います。yes を指定すると、受信した鍵が未知の場合は処理を中断します。no を指定すると、受信した鍵が未知の場合に確認を行わずに受けれて処理を継続します。

--

例 :

StrictHostKeyChecking no

--

2.2.21 LocalPasswordAuthentication

=====

対応 OS : Linux / Windows

書式 : LocalPasswordAuthentication <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no

=====

LPA 認証の設定を行います。no を指定した場合、サーバから認証を要求された際に LPA 認証を行いません。

--

例 :

LocalPasswordAuthentication no

--

2.2.22 PAMAuthentication

=====

対応 OS : Linux / Windows

書式 : PAMAuthentication <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no

=====

PAM 認証の設定を行います。no を指定した場合、サーバから認証を要求された際に PAM 認証を行いません。

--

例 :

PAMAuthentication no

--

2.2.23 PubkeyAuthentication

=====

対応 OS : Linux / Windows

書式 : PubkeyAuthentication <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no

=====

RSA 認証の設定を行います。no を指定した場合、サーバから認証を要求された際に RSA 認証を行いません。

--

例 :

PubkeyAuthentication no

--

2.2.24 WinLogonUserAuthentication

=====

対応 OS : Linux / Windows

書式 : WinLogonUserAuthentication <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no

=====

Windows 認証の設定を行います。no を指定した場合、サーバから認証を要求された際に Windows 認証を行いません。

--

例 :

WinLogonUserAuthentication no

--

2.2.25 PrivateKeySearchDir

=====

対応 OS : Linux / Windows

書式 : PrivateKeySearchDir <flag-available>

flag-available

既定値 :

/etc/hcp/keys (Linux)

C:/ProgramData/Clealink/HCP Tools/keys (Windows)

値の範囲 : ファイルシステムのパス文字列

=====

RSA 認証で使用するユーザの秘密鍵を特定するために探索するディレクトリを指定します。

--

例 :

PrivateKeySearchDir /etc/hcp/keys

--

指定されたディレクトリで次の様なファイル名を秘密鍵が保管されたファイルと見做して探索します。

<ユーザ名>.key

このユーザ名は、接続先のサーバのユーザ名 (-u オプションで指定するユーザ名もしくはログインプロンプトで入力するユーザ名) ではなく、ローカルコンピュータのユーザ名が使用されます。ローカルコンピュータのユーザ名がサーバのユーザ名と異なる場合はご注意ください。

秘密鍵と同じディレクトリに次の拡張子を持つ同じユーザ名のファイルが存在する場合は、クライアント証明書と見做して認証を行います。

- crt

- cer

秘密鍵は PEM 形式、OpenSSH 形式及び PuTTY 形式をサポートします。証明書は PEM 形式をサポートします。

2.2.26 PrivateKeyFile

=====

対応 OS : Linux / Windows

書式 : PrivateKeyFile <file-path>

file-path

既定値 :

~/.hcp/id_rsa (Linux)

~/_hcp/id_rsa (Windows)

値の範囲 : ユーザディレクトリを表すパス表記(~、チルダ)を含むファイルパス

=====

RSA 認証で使用するユーザの秘密鍵を特定するために探索するユーザホームディレクトリ内の鍵保管ファイルのパスを指定します。

```
--
例 :
PrivateKeyFile ~/.hcp/id_rsa
--
```

特定したファイルに次の拡張子が付加されたファイルが存在する場合は、クライアント証明書と見做して認証を行います。

```
- crt
- cer
```

秘密鍵は PEM 形式、OpenSSH 形式及び PuTTY 形式をサポートします。証明書は PEM 形式をサポートします。

2.2.27 AcceptableCryptMethod

=====
対応 OS : Linux / Windows

書式 : AcceptableCryptMethod <method-names>

method-names

書式 : <method-name>[...]

既定値 : AES256/CTR/VMAC AES256/CBC AES128/CBC

method-name

値の範囲 : PLAIN, AES128/CBC, AES192/CBC, AES256/CBC, AES128/CBC/HMAC, AES192/CBC/HMAC, AES256/CBC/HMAC, AES128/CBC/VMAC, AES192/CBC/VMAC, AES256/CBC/VMAC, AES128/CBC/VMAC64, AES192/CBC/VMAC64, AES256/CBC/VMAC64, AES128/CTR/HMAC, AES192/CTR/HMAC, AES256/CTR/HMAC, AES128/CTR/VMAC, AES192/CTR/VMAC, AES256/CTR/VMAC, AES128/CTR/VMAC64, AES192/CTR/VMAC64, AES256/CTR/VMAC64, AES128/GCM (予約), AES192/GCM (予約), AES256/GCM (予約)

=====
暗号アルゴリズムを設定します。

AES128/CBC と指定した場合は、AES128/CBC/HMAC と解釈されます（これらは同一のアルゴリズムです。AES192/CBC 及び AES256/CBC についても同様）。

新たに追加したアルゴリズム（CTR モード、VMAC モードを含むアルゴリズム）に対応していないバージョン（1.2.5 以前）と通信する場合は、これらの新しいアルゴリズムは接続時のネゴシエーションで無視されます（エラーとはなりません）。

--

例：

`AcceptableCryptMethod AES256/CBC PLAIN`

--

サーバとの間で通信するメッセージの暗号化に使用されます。どのアルゴリズムが使用されるかは、サーバの設定と併せて評価し、一致したアルゴリズムを使用します。

2.2.28 AcceptableDigestMethod

=====
対応 OS : Linux / Windows

書式 : `AcceptableDigestMethod <method-names>`

method-names

書式 : `<method-name>[...]`

既定値 : MM128 SHA256 SHA160

method-name

値の範囲 : NONE, SHA160, SHA224, SHA256, SHA384, SHA512, MD5, MM32, MM128

=====

ダイジェストアルゴリズムを設定します。

--

例：

`AcceptableDigestMethod SHA256 MM128 NONE`

--

サーバとの間で通信するメッセージ、ファイルやそのデータブロックの検証に使用されます。どのアルゴリズムが使用されるかは、サーバの設定と併せて評価し、一致したアルゴリズムを使用します。

また、データ検査に HMAC を使用する暗号通信 (AES256/CBC/HMAC など) を行う場合、セキュリティダイジェストではないアルゴリズム (MD5, MM32, MM128) は、指定されなかったものとして扱われます。

2.2.29 DisableDataIntegrityChecking

=====
対応 OS : Linux / Windows

書式 : `DisableDataIntegrityChecking <flag-available>`

flag-available

既定値 : no

値の範囲 : yes, no

サーバとの間で暗号化通信を行う場合に、MAC による通信メッセージの検査（データ完全性検査）を無効にする（省略する）ことをサーバに要求するか設定します。

yes を指定すると、サーバが許可した場合に通信メッセージの検査を行わない暗号化通信を行います。サーバが拒否した場合は、後述の設定項目

AcceptDataIntegrityCheckingOnRejection の設定に従って動作します。

通常は no（デフォルト）で使用してください。前述の通信メッセージの検査が省略されることを理解したうえでご利用ください。通常、暗号通信のパフォーマンス向上の目的で使用します。

```
--
例 :
DisableDataIntegrityChecking yes
--
```

2.2.30 AcceptDataIntegrityCheckingOnRejection

対応 OS : Linux / Windows

書式 : AcceptDataIntegrityCheckingOnRejection <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no

サーバとの間で暗号化通信を行う場合に、MAC による通信メッセージの検査を無効にする（省略する）要求が拒否された場合に、通信を継続か中断するか設定します。

yes を指定すると、通信メッセージの検査を行って通信を継続します。no を指定すると、通信を中断してアプリケーションを終了します。

```
--
例 :
AcceptDataIntegrityCheckingOnRejection no
--
```

2.2.31 TransportCharEncoding

対応 OS : Linux / Windows

書式 : TransportCharEncoding <encodings>

encodings

書式 : <encoding>[...]

既定値 : UTF8

encoding

値の範囲 : US-ASCII, UTF8, UTF16, UTF32

トランスポートで使用する文字列のエンコーディング方式を指定します。

--
例 :

TransportCharEncoding UTF8 UTF16 US-ASCII

--
サーバとの間で交換するファイルパスなどの文字列に適用されます。どのエンコーディングが使用されるかは、サーバの設定と併せて評価し、一致したエンコーディングを使用します。

2.2.32 HostEncoding

=====
対応 OS : Linux / Windows

書式 : HostEncoding <encoding>

encoding

既定値 :

UTF-8 (Linux)

CP932 (Windows)

値の範囲 : システムかつエンコーディング変換ライブラリでサポートされるエンコーディング名 (プラットフォーム依存)

=====
ホストで使用されている文字列エンコーディングを指定します。

--
例 :

HostEncoding EUC-JP

2.2.33 CompressLevel

=====
対応 OS : Linux / Windows

書式 : CompressLevel <compress-level>

compress-level

既定値 : -1

値の範囲 : -1, 0 - 9

=====

通信メッセージを圧縮するレベルを指定します。

--

例 :

CompressLevel 9

--

-1 を指定した場合、内部的にレベル 6 が選択されます。

0 を指定した場合、圧縮は行いません。

2.2.34 HeaderCompress

=====

対応 OS : Linux / Windows

書式 : HeaderCompress <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no

=====

ファイル要求などのヘッダ情報を圧縮する設定を行います。

--

例 :

HeaderCompress no

--

2.2.35 ContentCompress

=====

対応 OS : Linux / Windows

書式 : ContentCompress <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no

=====

ファイルのデータを圧縮する設定を行います。

--

例 :

ContentCompress no

--

2.2.36 MaxConcurrentThread

=====
対応 OS : Linux

書式 : MaxConcurrentThread <max-threads>

max-threads

既定値 : 0

値の範囲 : 符号付整数
=====

スレッド数制限を設定します。

2.2.37 MaxReceiveFileSize

=====
対応 OS : Linux / Windows

書式 : MaxReceiveFileSize <file-size>

file-size

既定値 : 8EB - 1B (符号付倍長整数最大値。制限なし)

値の範囲 : 符号付き倍長整数
=====

受信を許可する最大のファイルサイズを設定します。

--

例 :

MaxReceiveFileSize 1GB

--

2.2.38 MaxSendFileSize

=====
対応 OS : Linux / Windows

書式 : MaxSendFileSize <file-size>

file-size

既定値 : 8EB - 1B (符号付倍長整数最大値。制限なし)

値の範囲 : 符号付き倍長整数
=====

送信を許可する最大のファイルサイズを設定します。

--

例 :

MaxSendFileSize 1GB

--

2.2.39 MaxRequestFileEntryAtOnce

=====
対応 OS : Linux / Windows

書式 : MaxRequestFileEntryAtOnce <max-file-req-at-once>

max-file-req-at-once

既定値 : 50

値の範囲 : 符号付き整数
=====

ファイル要求の一括送信を許可する最大の数を設定します。

--

例 :

MaxRequestFileEntryAtOnce 1000

--

2.2.40 MaxBufferSize

=====
対応 OS : Linux / Windows

書式 : MaxBufferSize <max-buf-size>

max-buf-size

既定値 : 1GB

値の範囲 : 符号なし倍長整数
=====

ファイルのデータ処理に使用できる最大のメモリバッファサイズを設定します。

--

例 :

MaxBufferSize 1GB

--

2.2.41 MaxReceiveRate

=====
対応 OS : Linux / Windows

書式 : MaxReceiveRate <bandwidth>

bandwidth

既定値 : 10Gbit

値の範囲 : 符号なし倍長整数
=====

トランスポートの受信帯域のシェーピング（制限）を設定します。

```
--  
例 :  
MaxReceiveRate 1Gbit  
--
```

2.2.42 MaxSendRate

```
=====
```

対応 OS : Linux / Windows
書式 : MaxSendRate <bandwidth>

```
-----
```

bandwidth
既定値 : 10Gbit
値の範囲 : 符号なし倍長整数

```
=====
```

トランスポートの送信帯域のシェーピング（制限）を設定します。

```
--  
例 :  
MaxSendRate 1Gbit  
--
```

2.2.43 InitHeaderBlockSize

```
=====
```

対応 OS : Linux / Windows
書式 : InitHeaderBlockSize <block-size>

```
-----
```

block-size
既定値 : 50KB
値の範囲 : 符号なし倍長整数

```
=====
```

初期ヘッダブロックのサイズを設定します。

```
--  
例 :  
InitHeaderBlockSize 10KB  
--
```

ファイル要求などのメッセージを複数含むヘッダブロックが形成可能な上限のサイズを与えます。このオプションは通信開始直後に適用されます。

2.2.44 InitContentSize

```
=====
対応 OS : Linux / Windows
書式 : InitContentSize <block-size>
-----
block-size
既定値 : 1MB
値の範囲 : 符号なし倍長整数
=====
```

初期コンテンツブロックのサイズを設定します。

```
--
例 :
InitContentSize 2MB
--
```

ファイルのデータを複数含むコンテンツブロックが形成可能な上限のサイズを与えます。このオプションは通信開始直後に適用されます。

2.2.45 MaxHeaderBlockSize

```
=====
対応 OS : Linux / Windows
書式 : MaxHeaderBlockSize <block-size>
-----
block-size
既定値 : 50KB
値の範囲 : 符号なし倍長整数
=====
```

ヘッダブロックのサイズを拡張する最大のサイズを設定します。

```
--
例 :
MaxHeaderBlockSize 100KB
--
```

通信を開始すると消費帯域を評価して形成可能なヘッダブロックのサイズを増減させます。本オプションは、このサイズを増加させることができる上限値を与えます。

2.2.46 MaxContentSize

```
=====
対応 OS : Linux / Windows
書式 : MaxContentSize <block-size>
```

block-size

既定値 : 1MB

値の範囲 : 符号なし倍長整数

コンテンツブロックのサイズを拡張する最大のサイズを設定します。

10Gbps を超える環境などで通信性能が頭打ちになった場合などに、
InitContentBlockSize と併せて変更すると性能が改善する場合があります。

--

例 :

MaxContentBlockSize 4MB

--

通信を開始すると消費帯域を評価して形成可能なコンテンツブロックのサイズを増減させます。本オプションは、このサイズを増加させることができる上限値を与えます。

2.2.47 FileLock

hcpd 設定項目参照

2.2.48 FileLockTrials

hcpd 設定項目参照

2.2.49 FileLockTrialInterval

hcpd 設定項目参照

2.2.50 AtomicLikeSaving

hcpd 設定項目参照

2.2.51 AtomicLikeSavingThreshold

hcpd 設定項目参照

2.2.52 TransportTimeout

hcpd 設定項目参照

2.2.53 ApplicationLog

対応 OS : Linux / Windows

書式 : ApplicationLog <log-level>[<flag-available>][<log-rotation-conf>][<log-path>]

log-level

既定値 : INFO

値の範囲 : EMERG, ALERT, CRIT, ERR, WARNING, INFO, DEBUG

flag-available

既定値 : no

値の範囲 : yes, no

log-rotation-conf

書式 : (FileSize <file-size> <backups> | DatePattern <date-pattern>)

file-size

既定値 : なし

値の範囲 : 符号あり倍長整数

backups

既定値 : なし

値の範囲 : 符号なし整数

date-pattern

既定値 : なし

値の範囲 : yyyy-MM, yyyy-MM-dd, yyyy-MM-dd-HH, yyyy-MM-dd-HH-mm

log-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

=====

アプリケーションのログ設定を行います。

log-level は、ログのレベルを指定します。

flag-available に **yes** を指定するとログを出力します。

log-rotation-conf は、ログのローテーションを指定します。**SystemLog** で設定するローテーションと同様の動作を行います。但し、定期的なローテーションは行いません。

log-path は、ログのパスを指定します。コマンドラインパラメータの **-l** とともに設定した場合は、コマンドラインパラメータの指定が適用されます。

```
--
例 1 :
ApplicationLog WARNING FileSize 10MB 10
例 2 :
ApplicationLog WARNING DatePattern yyyy-MM-dd
例 3 :
ApplicationLog WARNING // ApplicationLogLevel に同じ
--
```

2.2.54 ApplicationLogLevel

```
=====
対応 OS : Linux / Windows
書式 : ApplicationLogLevel <log-level>
-----
log-level
既定値 : INFO
値の範囲 : EMERG, ALERT, CRIT, ERR, WARNING, INFO, DEBUG
=====
```

アプリケーションのログレベルを設定します。

ApplicationLog と共に記述した場合は、ログレベルの値のみ上書きされます。

```
--
例 :
ApplicationLogLevel WARNING
--
```

2.2.55 ApplicationStatLog

```
=====
対応 OS : Linux / Windows
書式 : ApplicationStatLog <flag-available>[ <log-rotation-conf>]
-----
flag-available
既定値 : yes
値の範囲 : yes, no
-----
log-rotation-conf
書式 : ( FileSize <file-size> <backups> | DatePattern <date-pattern> )
-----
file-size
既定値 : なし
値の範囲 : 符号あり倍長整数
-----
backups
```

既定値 : なし
値の範囲 : 符号なし整数

date-pattern

既定値 : なし
値の範囲 : yyyy-MM, yyyy-MM-dd, yyyy-MM-dd-HH, yyyy-MM-dd-HH-mm

=====

アプリケーション統計を設定します。

flag-available に yes を指定するとアプリケーションの統計情報を出力します。

log-rotation-conf は、出力のローテーションを指定します。hcpd の設定項目 ApplicationStatLog と同様の動作を行います。

--
例 1 :
ApplicationStatLog no
例 2 :
ApplicationStatLog yes FileSize 10MB 10
例 3 :
ApplicationStatLog yes DatePattern yyyy-MM-dd
--

2.2.56 TransportStatLog

=====

対応 OS : Linux / Windows

書式 : TransportStatLog <flag-available>[<log-rotation-conf>]

flag-available

既定値 : no
値の範囲 : yes, no

log-rotation-conf

書式 : (FileSize <file-size> <backups> | DatePattern <date-pattern>)

file-size

既定値 : なし
値の範囲 : 符号あり倍長整数

backups

既定値 : なし
値の範囲 : 符号なし整数

date-pattern

既定値 : なし

値の範囲 : yyyy-MM, yyyy-MM-dd, yyyy-MM-dd-HH, yyyy-MM-dd-HH-mm

=====

トランスポート統計を設定します。

flag-available に yes を指定するとトランスポートの統計情報を出力します。

log-rotation-conf は、出力のローテーションを指定します。hcpd の設定項目 TransportStatLog と同様の動作を行います。

FileSize と DatePattern でそれぞれ統計ログの基準パスとして指定されたパスをもとに次の様にローテーションが行われます。

```
// FileSize の場合
<指定されたパス>.transport.tcp
<指定されたパス>.transport.tcp.1
<指定されたパス>.transport.tcp.2
...
<指定されたパス>.transport.tcp.n
// DatePattern の場合
<指定されたパス>.transport.tcp
<指定されたパス>.transport.tcp.2019-12-10
<指定されたパス>.transport.tcp.2019-12-09
...
```

ローテーションされたファイルに統計情報のヘッダは含まれません。

--

例 1 :

TransportStatLog yes

例 2 :

TransportStatLog yes FileSize 10MB 10

例 3 :

TransportStatLog yes DatePattern yyyy-MM-dd

--

2.2.57 UDPTransportExtensionBufferSize

=====

対応 OS : Linux / Windows

書式 : UDPTransportExtensionBufferSize <ext-buf-size>

ext-buf-size

既定値 : 2GB

値の範囲 : 符号なし倍長整数 (バイト単位)

=====

HpFP(UDP)トランスポートで使用する拡張バッファのサイズを指定します。

HpFP セッションでは、遅延やロス及び通信量の増加などに伴い通信用のバッファを指定されたサイズ (UDPListenAddress の `hpfp_sndbuf` 及び `hpfp_rcvbuf`) に拡張します。この拡張するバッファの合計容量を指定された値で制限します。

0 を指定した場合は、この制限を行いません。

また、バッファの初期サイズ (拡張される前のバッファのサイズ) は 1MB です。

--
例 :
UDPTransportExtensionBufferSize 4GB
--

2.2.58 TCPTransportSocketSendBuffer

=====

対応 OS : Linux / Windows
書式 : TCPTransportSocketSendBuffer <snd-buf-size>

snd-buf-size
書式 : <decimal_number>[[(T|G|M|K)]B]
既定値 : 0
値の範囲 : 符号なし倍長整数 (バイト単位)

=====

TCP 通信で指定する送信バッファのサイズ (バイト単位) を設定します。0 を指定すると未指定となります。

100G 環境などで TCP 通信性能のチューニングが必要な場合などに使用します。通常は変更する必要はありません。

--
例 :
TCPTransportSocketSendBuffer 128MB
--

2.2.59 MemoryTransferConcurrency

=====

対応 OS : Linux / Windows
書式 : MemoryTransferConcurrency <num-concur> <wait-type> <busy-sleep-nsec>

num-concur

既定値 : 自動 (物理コア数 / 2 と 16 のいずれか小さい値)

値の範囲 : 符号なし整数

wait-type

既定値 : cond

値の範囲 : cond, busy

busy-sleep-nsec

既定値 : 1

値の範囲 : 符号なし整数

ローカル I/O を抑制するモード (hcp コマンド -n オプション。メモリツーメモリ転送動作時) を実行する場合に、送信元で使用する疑似データをデータブロックバッファにコピー (memcpy によるメモリコピー) する際の並列動作を設定します。

num-concur は、並列数を指定します。1 を指定すると並列化せずに標準のメモリコピー処理(memcpy)を実行します。

wait-type は、並列動作するメモリコピースレッドが待機する際に使用する待機方式を指定します。cond は条件付き待機(conditional wait)を、busy はビジー待機(busy wait)を行います。busy を指定した場合は、並列数を指定した数の CPU の使用率が常時 100%まで達する場合があります。

busy-sleep-nsec は、ビジー待機する際に待機する時間をナノ秒で指定します。

このオプションは 100G 環境などでベンチマークを行う際に、シングルスレッドによるメモリコピー(memcpy)の性能限界 (このメモリコピーのボトルネックがアプリ通信性能のボトルネックになる) を解消するための性能チューニングに使用します。

--

例 :

MemoryTransferConcurrency 1 cond 1 # 無効化

MemoryTransferConcurrency 12 busy 1 # ビジー待機、4 並列、1 ナノ秒待機

--

2.2.60 MaxReadRate

対応 OS : Linux / Windows

書式 : MaxReadRate <read-rate>

read-rate

既定値 : 10Ebit (無制限)

値の範囲 : 符号なし倍長整数 (10Ebit まで)

=====

ファイル転送時にディスクからファイルを読み込む速度の上限を設定します。この設定はファイル転送のセッション毎に適用されます。bps 単位で符号付倍長整数の最大値もしくはそれを超える値が指定された場合は無制限と設定されます。

この設定は SSD などを使用する際に長時間の読み書きを高レートで行うとオーバーヒートを起こすことがあります (特に書込み側)、それを抑制する場合などに使用します。

--

例 :

MaxReadRate 10Gbit

--

2.2.61 MaxWriteRate

=====

対応 OS : Linux / Windows

書式 : MaxWriteRate <write-rate>

write-rate

既定値 : 10Ebit (無制限)

値の範囲 : 符号なし倍長整数 (10Ebit まで)

=====

ファイル転送時にディスクからファイルを書込む速度の上限を設定します。この設定はファイル転送のセッション毎に適用されます。bps 単位で符号付倍長整数の最大値もしくはそれを超える値が指定された場合は無制限と設定されます。

この設定は SSD などを使用する際に長時間の読み書きを高レートで行うとオーバーヒートを起こすことがあります (特に書込み側)、それを抑制する場合などに使用します。

--

例 :

MaxWriteRate 10Gbit

--

2.2.62 PubkeyAuthenticationPrior

=====

対応 OS : Linux / Windows

書式 : PubkeyAuthenticationPrior <flag-available>

```
-----
flag-available
```

```
既定値 : yes
```

```
値の範囲 : yes, no
=====
```

RSA 認証（公開鍵方式認証）を優先して実行するか指定します。

yes の場合、まず RSA 鍵の探索とロードを試みます。RSA 鍵が暗号化されている場合は、都度復号するためのパスワードの入力が要求されます（Windows 版は非サポート）。最初にロードに成功した RSA 鍵を使用して RSA 認証を行います（その他の RSA 鍵は無視されます）。RSA 鍵がロードできなかった場合は、パスワード認証が行われます。

no の場合は、従来の認証動作を行います。パスワードが未入力の場合は入力が必要され、全ての認証方式が試行されます。

```
--
```

```
例 :
```

```
PubkeyAuthenticationPrior no
```

```
--
```

2.2.63 ApplicationStatLogSecurityEx

```
=====
対応 OS : Linux / Windows
```

```
書式 : ApplicationStatLogSecurityEx <flag-available>
-----
```

```
flag-available
```

```
既定値 : yes
```

```
値の範囲 : yes, no
=====
```

アプリ統計ログでセキュリティに関する詳細情報を出力するか設定します。

```
--
```

```
例 :
```

```
ApplicationStatLogSecurityEx no
```

```
--
```

2.3 hcp.conf

項目名	説明
Include	外部設定ファイルのインクルード
UseProperCopyAndSync	cp/rsync に近いコピー（同期）動作の指定

AutoResumeTrials	自動再開試行回数
AutoResumeTrialInterval	自動再開試行間隔（秒）

2.3.1 Include

=====
 対応 OS : Linux / Windows
 書式 : Include <file-path>

file-path

既定値 : なし

値の範囲 : ファイルシステムのパス文字列

=====
 クライアントコマンドに共通して指定できる設定を記述したファイルをインクルードします。インクルードは、**hcp.conf** の任意の位置に記述できます。インクルードの記述の前に設定された同じ項目の値は上書きされます。インクルードはネストできません。

--
 例 :
 Include /etc/hcp/hcp-common.conf
 --

2.3.2 UseProperCopyAndSync

=====
 対応 OS : Linux / Windows
 書式 : UseProperCopyAndSync <flag-available>

flag-available

既定値 : yes

値の範囲 : yes, no

=====
 コピー動作及び同期動作をそれぞれ **cp** コマンド、**rsync** コマンドの動作に合わせるように指示をします。

コピー動作は、主に次の様に動作が変更されます。

- 転送元にディレクトリを指定し、かつ宛先のパスがディレクトリとして存在する場合は、宛先のディレクトリに転送元のディレクトリ名でディレクトリを作成し、その配下にファイルをコピーします

- 転送元にディレクトリを指定し、かつ宛先のパスが存在しない場合は、宛先のパス名でディレクトリを作成し、その配下にファイルをコピーします（転送元のディレクトリ名でディレクトリは作成されません）
- ファイルもしくはディレクトリの上書きでファイル種別が異なる場合（例：転送元はファイル、転送先はディレクトリ）は、理由コード(CANNOT_OVERWRITE_DIR, CANNOT_OVERWRITE_NON_DIR)でエラーになります
- 転送元にディレクトリを指定し、かつ-R（再帰コピー）オプションを指定しない場合は、理由コード(OMIT_DIR)でエラーになります
- 転送元に複数のパスを指定し、かつ宛先のパスがディレクトリでない場合は、理由コード(DEST_IS_NON_DIR)でコマンドの実行を停止します
- 転送元が複数のファイルを含む可能性があり（ディレクトリ、正規表現オプション指定、ワイルドカード）、かつ宛先のパスが存在しディレクトリでない場合は、理由コード(CANNOT_OVERWRITE_NON_DIR)でコマンドの実行を停止します

同期動作は、主に次の様に動作が変更されます。

- 転送元にディレクトリを指定し、かつこのパス指定の末尾がスラッシュ(/)でなければ、宛先パスのディレクトリ（存在しない場合は作成）に転送元のディレクトリ名でディレクトリを作成し、その配下にファイルをコピーします
- 転送元にディレクトリを指定し、かつこのパス指定の末尾がスラッシュ(/)の場合は、宛先パスのディレクトリ（存在しない場合は作成）の配下にファイルをコピーします
- ファイルもしくはディレクトリの上書きでファイル種別が異なる場合（例：転送元はファイル、転送先はディレクトリ）は、転送先のファイルもしくはディレクトリを削除してからコピーを行います
- 転送元にディレクトリを指定し、かつ-R（再帰コピー）オプションを指定しない場合は、理由コード(OMIT_DIR)で処理をスキップし、コマンドの実行結果はエラーとして評価しません（終了ステータスが 0 で返る）
- 転送元が複数のファイルを含む可能性があり（ディレクトリ、正規表現オプション指定、ワイルドカード）、かつ宛先のパスが存在しディレクトリでない場合は、理由コード(CANNOT_OVERWRITE_NON_DIR)でコマンドの実行を停止します

--

例：

UseProperCopyAndSync no

--

2.3.3 AutoResumeTrials

=====
 対応 OS : Linux.x86 / Windows
 書式 : AutoResumeTrials <num-trials>

num-trials
 既定値 : -1
 値の範囲 : 符号付き整数
 =====

再開を自動的に試行する回数を設定します。-1 を指定すると回数の制限なく再開を試行します。0 を指定した場合、再開を行わずに停止します（再開試行回数 0 回）。

--
 例 :
 AutoResumeTrials 5
 --

2.3.4 AutoResumeTrialInterval

=====
 対応 OS : Linux.x86 / Windows
 書式 : AutoResumeTrialInterval <trial-interval>

trial-interval
 既定値 : 3
 値の範囲 : 符号なし整数
 =====

再開を自動的に試行するまでの待機時間間隔を設定します（秒単位）。

--
 例 :
 AutoResumeTrialInterval 10
 --

2.4 hrm.conf

項目名	説明
Include	外部設定ファイルのインクルード

2.4.1 Include

hcp.conf 参照

2.5 hcp-ls.conf

項目名	説明
Include	外部設定ファイルのインクルード

2.5.1 Include

hcp.conf 参照

2.6 hmkdir.conf

項目名	説明
Include	外部設定ファイルのインクルード

2.6.1 Include

hcp.conf 参照

2.7 hpwd.conf

項目名	説明
Include	外部設定ファイルのインクルード

2.7.1 Include

hcp.conf 参照

2.8 hmv.conf

項目名	説明
Include	外部設定ファイルのインクルード

2.8.1 Include

hcp.conf 参照

2.9 hln.conf

項目名	説明
Include	外部設定ファイルのインクルード

2.9.1 Include

hcp.conf 参照

2.10 users

hcpd デーモンが認証で認識するユーザを定義します。各行に次の書式でユーザを定義します。

書式 [!]<username>:<auth_methods>:[<mapping_uid>]:[<mapping_gid>]:[<hpfp_cong_modes>]:[<home_dir>]

username は、ユーザ名を記述します。セパレータを除く文字が使用可能で、システム上にユーザとして定義されない文字列（HCP tools 固有に定めたユーザ名）を指定することもできます。先頭にエクスクラメーション・マーク"!"を記述するとユーザの認証を拒否します。

auth_methods は、このユーザで利用を許可する認証方法をカンマ(,)区切りで列挙します。使用できる値は、次の通りです。

- LPA
- PAM
- RSA
- WLU

mapping_uid は、このユーザに指定したい **UID** を記述します。省略された場合は、ユーザ名からシステムへ問い合わせを行います。

mapping_gid は、このユーザに指定したい **GID** を記述します。

hpfp_cong_modes は、このユーザで許可する HpFP トランスポートの輻輳制御モードを 0 個以上指定します。**MODEST**、**FAIR_FAST_START** 及び **AGGRESSIVE** の三つのモードの省略表記（それぞれ、**M**、**S** 及び **A**）を指定します。**FAIR** モードはデフォルトで含まれます。

home_dir は、このユーザのホームディレクトリを記述します。省略された場合は、ユーザ名からシステムへ問い合わせを行います。

--

例：

```
hcp_user:LPA,RSA:1000:1000::/home/user
user:PAM:::
```

--

2.11 passwd

hcpd デーモンが **LPA** 認証で使用するユーザの資格情報（ハッシュ化されたパスワード）を定義します。各行に次の書式で資格情報を定義します。

書式 <username>:<hash_method>:<hash_value>

username は、ユーザ名を記述します。

hash_method は、資格情報の生成方法を次の何れかを記述します。

- md5
- sha
- sha224
- sha256
- sha384
- sha512

hash_value は、次の式に従って変換されたハッシュ化された文字列を記述します。

hex_string (hash_function (<ユーザ入力パスワード>))

hex_string は、バイナリデータを 16 進数文字列表記に変換する関数です。

hash_function は、hash_method で指定したダイジェスト関数です。

通常この文字列は、下記の様にオペレーティングシステムのツールを使用して作成することができます。

```
$ echo -n my_password | openssl dgst -md5
(stdin)= a865a7e...93bea4
```

--

例：

```
# password is "password01"
hcp_user:sha256:4b8f353889d9a05d17946e26d014efe99407cba8bd9d0102d4aab10ce
6229043
```

--

3 統計リファレンス

3.1 アプリケーション統計

ファイル転送(hcp)やファイル削除(hrm)等の各アプリケーション実行後に、クライアントおよびサーバ上でその実行結果の情報を所定のファイルに記録します。

列名	説明
Start Date_Time	開始日時

End Date_Time	終了日時
Exit	終了コード
Remote Host	リモートホスト
User	ユーザ名
Application	アプリケーション名
Total	トータルデータ伝送量(バイト)
Payload	ファイルデータ伝送量(バイト)
Files	ファイル数
Directories	ディレクトリ数
Average Throughput	平均スループット(bps)
Encoding	トランスポート文字エンコーディング
Security	トランスポートセキュリティ
Compression	圧縮
Proto	トランスポートプロトコル
Local	ローカルアドレス及びポート番号
Foreign	リモートアドレス及びポート番号
PID	プロセス ID

Start Date_Time は、アプリケーションの実行開始日時が記録されます。次の書式で出力されます。

yyyy/mm/dd HH:MM::SS

End Date_Time は、アプリケーションの実行終了日時が記録されます。次の書式で出力されます。

yyyy/mm/dd HH:MM::SS

Exit は、アプリケーションの終了コードとしてファイル処理理由コード（後述）が記録されます。

Remote Host は、リモート（対向）のホストのホスト名もしくは IP アドレスが記録されます。

User は、アプリケーションを実行したユーザ及び認証方法(PAM など)が記録されます。認証を行わなかった場合は、anonymous が出力されます。

Application は、この記録を生成したアプリケーションの名前が記録されます。次の何れかが出力されます。

- hcp
- hsync
- hrm
- hcp-ls
- hmkdir
- hpwd
- hmv
- hln
- hchmod
- hchown

また、アプリケーションを実行したプロトコル（通信手順）のバージョンが括弧内に下記の様に記載されます（順にプロトコル番号、プロトコル改訂番号）。

..., hcp [2.0], ...

Total は、アプリケーション層でのデータ転送のために必要となった全バイト数（ファイル要求等の制御に必要なデータ量を含む）が記録されます。

Payload は、アプリケーション層のデータを伝送した量（ファイル転送の場合、ファイルのデータサイズの合計）が記録されます。

Files は、アプリケーション層で処理されたファイル数が記録されます（ディレクトリは含みません）。

Directories は、アプリケーション層で処理されたディレクトリ数が記録されます。

Average Throughput は、アプリケーション層のデータの平均伝送スループットが記録されます。

Encoding は、使用されたトランスポート文字エンコーディングが記録されます。

Security は、使用されたトランスポートセキュリティが記録されます。ダイジェストアルゴリズムの用途が次の名称で付記されます。

- MAC (セッション暗号時のメッセージデータの検査)
- Chunk (ファイルのデータチャンクの検査)
- File (ファイルの検査)

セッション暗号時において、次の設定によりメッセージデータの検査が無効にされた場合は、“MAC”は付記されません。

- RequireDataIntegrityChecking (hcpd.conf)
- DisableDataIntegrityChecking (hcp.conf)

ApplicationStatLogSecurityEx が yes の場合は、次のトランスポートセキュリティに関連する情報が追加で記録されます。

- Bxxx (暗号アルゴリズムブロックサイズ。xxx はブロックのサイズ。ビット単位)
- Kxxx (暗号アルゴリズム鍵サイズ。xxx は鍵のサイズ。ビット単位)
- HmacXxx/VmacXxx (データ完全性検査モード。Xxx はアルゴリズム名)

Compression は、使用された圧縮モード（圧縮レベル、ヘッダ圧縮およびコンテンツ圧縮の ON/OFF）が記録されます。

Proto は、使用されたトランスポートプロトコルが記録されます。現在は、次の何れかが出力されます。

- TCP
- HpFP
- WSS
- WS

HpFP が記録される場合は、選択された輻輳制御モードが次の名称で付記されます。

- F (Fair モード)
- S (Fair Fast モード)
- A (Aggressive モード)
- M (Modest モード)
- N (指定なし)

多重接続が行われた場合は、確立された接続数が括弧で付記されます。

Local は、トランスポートの通信セッションのローカル側の端点（エンドポイント）の情報（IP アドレス及びポート番号）が記録されます。

Foreign は、トランスポートの通信セッションのリモート側の端点（エンドポイント）の情報（IP アドレス及びポート番号）が記録されます。

PID は、アプリケーションのプロセス ID が記録されます。

--

例：

クライアント

```
Start Date_Time, End Date_Time, Exit, Remote Host, User, Application, Total (bytes), Payload (bytes), Files, Directories, Average Throughput (bps), Encoding, Security, Compression, Proto, Local, Foreign, PID
2021/01/06 16:27:24, 2021/01/06 16:27:30, 0000, 127.0.0.1, user, hcp [2.0], 8590723508, 8589934592, 1, 0, 10703091712.000000, UTF8, AES/CTR/NoPadding SHA256 [MAC / B128 K256 VmacAES64], None, TCP, 127.0.0.1:40920, 127.0.0.1:874, 74612
2021/02/02 14:55:36, 2021/02/02 14:55:36, 0000, 192.168.30.52, user, hcp [2.0], 1053400, 1048641, 1, 0, 21563498.000000, UTF8, AES/CBC/PKCS5Padding SHA256 [MAC / B128 K256 HmacSHA256], None, TCP, 192.168.30.51:35858, 192.168.30.52:874, 3651
2021/02/02 15:28:24, 2021/02/02 15:28:24, 0000, 192.168.30.52, user, hcp [2.0], 1052440, 1048641, 1, 0, 21731128.000000, UTF8, AES/CBC/PKCS5Padding SHA256 [B128 K256], None, TCP, 192.168.30.51:35890, 192.168.30.52:874, 3832
2021/02/02 15:10:50, 2021/02/02 15:10:50, 0000, 192.168.30.52, user, hcp [2.0], 1053400, 1048641, 1, 0, 21753950.000000, UTF8, AES/CBC/PKCS5Padding SHA256 [MAC / B128 K128 HmacSHA256], None, TCP, 192.168.30.51:35864, 192.168.30.52:874, 3727
2021/02/02 15:19:57, 2021/02/02 15:19:58, 0000, 192.168.30.52, user, hcp [2.0], 1050772, 1048641, 1, 0, 19767822.000000, UTF8, None SHA256, None, TCP, 192.168.30.51:35880, 192.168.30.52:874, 3794
2021/02/02 15:29:37, 2021/02/02 15:19:58, 0000, 192.168.30.52, user, hcp [2.0], 1050772, 1048641, 1, 0, 19767822.000000, UTF8, None SHA256, None, TCP (8), 192.168.30.51:35880, 192.168.30.52:874, 3794
2021/02/02 17:31:25, 2021/02/02 17:31:25, 0000, 127.0.0.1, user, hcp [2.0], 2160, 0, 1, 0, 0.000000, UTF8, AES/CTR/NoPadding SHA256 [MAC], None, HpFP [N], 127.0.0.1:54918, 127.0.0.1:884, 41340
2021/02/02 17:32:28, 2021/02/02 17:32:28, 0000, 127.0.0.1, user, hcp [2.0], 2160, 0, 1, 0, 0.000000, UTF8, AES/CTR/NoPadding SHA256 [MAC / B128 K256 VmacAES], None, HpFP [N], 127.0.0.1:21373, 127.0.0.1:884, 41458
2021/02/02 17:33:41, 2021/02/02 17:33:41, 0000, 127.0.0.1, user, hcp [2.0], 2480, 0, 1, 0, 0.000000, UTF8, AES/CBC/PKCS5Padding SHA256 [MAC / B128 K256 HmacSHA256], None, HpFP [N], 127.0.0.1:50812, 127.0.0.1:884, 41556
2021/02/02 17:43:21, 2021/02/02 17:33:41, 0000, 127.0.0.1, user, hcp [2.0], 2480, 0, 1, 0, 0.000000, UTF8, AES/CBC/PKCS5Padding SHA256 [MAC / B128 K256 HmacSHA256], None, HpFP [N] (8), 127.0.0.1:50812, 127.0.0.1:884, 41556
```

サーバ

```
Start Date_Time, End Date_Time, Exit, Remote Host, User, Application, Total (bytes), Payload (bytes), Files, Directories, Average Throughput (bps), Encoding, Security, Compression, Proto, Local, Foreign, PID
```

```

2021/01/08 14:35:45, 2021/01/08 14:35:45, 0000, 127.0.0.1:37100, user [RS
A], hcp [2.0], 3428, 1024, 1, 0, 20145.384766, UTF8, AES/CTR/NoPadding SH
A256 [B128 K256], None, TCP, 127.0.0.1:874, 127.0.0.1:37100, 57562
2021/01/08 14:36:10, 2021/01/08 14:36:10, 0000, 127.0.0.1:37102, user [RS
A], hcp [2.0], 3428, 1024, 1, 0, 19958.435547, UTF8, AES/CTR/NoPadding SH
A256, None, TCP, 127.0.0.1:874, 127.0.0.1:37102, 57650
2021/01/08 15:26:30, 2021/01/08 14:36:10, 0000, 127.0.0.1:37102, user [RS
A], hcp [2.0], 3428, 1024, 1, 0, 19958.435547, UTF8, AES/CTR/NoPadding SH
A256, None, TCP (8), 127.0.0.1:874, 127.0.0.1:37102, 57650
2021/02/02 17:31:25, 2021/02/02 17:31:25, 0000, 127.0.0.1:54918, user [PA
M], hcp [2.0], 2160, 0, 1, 0, 0.000000, UTF8, AES/CTR/NoPadding SHA256 [M
AC], None, HpFP [N], 127.0.0.1:884, 127.0.0.1:54918, 41349
2021/02/02 17:32:28, 2021/02/02 17:32:28, 0000, 127.0.0.1:21373, user [PA
M], hcp [2.0], 2160, 0, 1, 0, 0.000000, UTF8, AES/CTR/NoPadding SHA256 [M
AC / B128 K256 VmacAES], None, HpFP [N], 127.0.0.1:884, 127.0.0.1:21373,
41465
2021/02/02 17:33:41, 2021/02/02 17:33:41, 0000, 127.0.0.1:50812, user [PA
M], hcp [2.0], 2480, 0, 1, 0, 0.000000, UTF8, AES/CBC/PKCS5Padding SHA256
[MAC / B128 K256 HmacSHA256], None, HpFP [N], 127.0.0.1:884, 127.0.0.1:5
0812, 41563
2021/02/02 17:43:21, 2021/02/02 17:33:41, 0000, 127.0.0.1:50812, user [PA
M], hcp [2.0], 2480, 0, 1, 0, 0.000000, UTF8, AES/CBC/PKCS5Padding SHA256
[MAC / B128 K256 HmacSHA256], None, HpFP [N] (8), 127.0.0.1:884, 127.0.0.
1:50812, 41563
--

```

3.2 トランスポート統計

3.2.1 TCP 統計

TCP による通信を実行中にサーバおよびクライアントで通信量やトランスポート統計情報を定期的（約 1 秒毎）に記録します。

列名	説明
Date_Time	記録日時
Tx	送信サイズ(バイト)
Rx	受信サイズ(バイト)
AcqTx	送信帯域獲得サイズ(バイト)
AcqRx	受信帯域獲得サイズ(バイト)
Cwnd	輻輳ウィンドウサイズ(バイト)
RTT	RTT (マイクロ秒)
Retrans	再送回数

TpTx	送信スループット(bps)
TpRx	受信スループット(bps)
AvgTpTx	送信平均スループット(bps)
AvgTpRx	受信平均スループット(bps)

Date_Time は、この記録が作成された日時が記録されます。次の書式で出力されます。

yyyy/mm/dd HH:MM::SS.aaaaaa (マイクロ秒まで記載)

Tx は、この記録が作成されるまでにトランスポート層で送信されたバイト数が記録されます (差分)。

Rx は、この記録が作成されるまでにトランスポート層で受信したバイト数が記録されます (差分)。

AcqTx は、この記録が作成されるまでにトランスポート層で獲得された送信帯域のバイト数が記録されます (差分)。

AcqRx は、この記録が作成されるまでにトランスポート層で獲得された受信帯域のバイト数が記録されます (差分)。

Cwnd は、この記録が作成された時点で TCP の統計情報から取得した輻輳ウィンドウのサイズが記録されます。

RTT は、この記録が作成された時点で TCP の統計情報から取得した RTT が記録されます。

Retrans は、この記録が作成された時点で TCP の統計情報から取得した再送カウントが記録されます。

TpTx は、この記録のトランスポート層の送信スループットが記録されます。

TpRx は、この記録のトランスポート層の受信スループットが記録されます。

AvgTpTx は、この記録が作成された時点でのトランスポート層の平均送信スループットが記録されます。

AvgTpRx は、この記録が作成された時点でのトランスポート層の平均受信スループットが記録されます。

--
例 :
--

```

PID 11168, Local 127.0.0.1:58374, Foreign 127.0.0.1:11111
Date_Time, Tx (bytes), Rx (bytes), AcqTx (bytes), AcqRx (bytes), Cwnd (bytes),
RTT (usec), Retrans, TpTx (bps), TpRx (bps), AvgTpTx (bps), AvgTpRx (bps)
2018/05/25 15:05:50.142156, 0, 0, 0, 0, 218450, 17, 0, 0.000000, 0.000000,
0.000000, 0.000000
2018/05/25 15:05:52.868790, 2288, 1760, 2288, 1764, 654830, 21, 0, 6712.9
42871, 5163.802246, 6712.854004, 5163.731934
2018/05/25 15:05:53.321617, 4436, 2920, 4436, 2920, 654830, 12764, 0, 783
69.882812, 51587.031250, 16918.314453, 11775.388672
--

```

3.2.2 HpFP 統計

HpFP による通信を実行中にサーバおよびクライアントで通信量やトランスポート統計情報を定期的（約 1 秒毎）に記録します。

列名	説明
Date_Time	記録日時
Tx	送信サイズ(バイト)
Rx	受信サイズ(バイト)
AcqTx	送信帯域獲得サイズ(バイト)
AcqRx	受信帯域獲得サイズ(バイト)
Cwnd	輻輳ウィンドウサイズ(バイト)
RTT	RTT (マイクロ秒)
Retrans	再送データサイズ(バイト)
Congestion	輻輳検出カウント
MSS	MSS (バイト)
MaxTp	最大スループット(bps)
LongCtxWait	コンテキスト長時間待機カウント
TpTx	送信スループット(bps)
TpRx	受信スループット(bps)
AvgTpTx	送信平均スループット(bps)
AvgTpRx	受信平均スループット(bps)

Date_Time は、この記録が作成された日時が記録されます。次の書式で出力されます。

yyyy/mm/dd HH:MM::SS.aaaaaa (マイクロ秒まで記載)

Tx は、この記録が作成されるまでにトランスポート層で送信されたバイト数が記録されます (差分)。

Rx は、この記録が作成されるまでにトランスポート層で受信したバイト数が記録されます (差分)。

AcqTx は、この記録が作成されるまでにトランスポート層で獲得された送信帯域のバイト数が記録されます (差分)。

AcqRx は、この記録が作成されるまでにトランスポート層で獲得された受信帯域のバイト数が記録されます (差分)。

Cwnd は、この記録が作成された時点で **HpFP** の統計情報から取得した輻輳ウィンドウのサイズが記録されます。

RTT は、この記録が作成された時点で **HpFP** の統計情報から取得した **RTT** が記録されます。

Retrans は、この記録が作成された時点で **HpFP** の統計情報から取得した再送カウントが記録されます。

Congestion は、この記録が作成された時点で **HpFP** の統計情報から取得した輻輳検出カウント (累積値) が記録されます。

MSS は、この記録が作成された時点で **HpFP** の統計情報から取得した **MSS** が記録されます。

MaxTp は、この記録が作成された時点で **HpFP** の統計情報から取得した最大スループットが記録されます。

LongCtxWait は、この記録が作成された時点で **HpFP** の統計情報から取得したコンテキスト長時間待機カウントが記録されます。このカウントが計上されている場合は、コンテキストスイッチによる性能への影響が発生している場合があります。

TpTx は、この記録のトランスポート層の送信スループットが記録されます。

TpTx は、この記録のトランスポート層の受信スループットが記録されます。

AvgTpTx は、この記録が作成された時点でのトランスポート層の平均送信スループットが記録されます。

AvgTpRx は、この記録が作成された時点でのトランスポート層の平均受信スループットが記録されます。

--

例 :

--

```

PID 2848, Local 127.0.0.1:63304, Foreign 127.0.0.1:884
Date_Time, Tx (bytes), Rx (bytes), TxAcq (bytes), RxAcq (bytes), Cwnd (bytes),
RTT (usec), Retrans (bytes), Congestion, MSS (bytes), MaxTp (bps),
LongCtxWait, TpTx (bps), TpRx (bps), AvgTpTx (bps), AvgTpRx (bps)
2019/10/30 09:34:17.520056, 0, 0, 0, 0, 89560, 382, 0, 0, 1236, 0, 0, 0.000000,
0.000000, 0.000000, 0.000000
2019/10/30 09:34:19.915566, 1052, 1124, 1052, 66536, 89560, 382, 0, 0, 1236,
5536, 0, 3511.808594, 3752.160400, 3511.798340, 3752.149414
2019/10/30 09:34:20.300687, 1708, 620, 1708, 744, 89560, 43, 0, 0, 1236,
85832, 0, 35479.757812, 12879.069336, 7937.694824, 5015.702637

```

--

3.2.3 WS/WSS 統計

WS (WebSocket) 及び WSS (WebSocketSecure) による通信を実行中にサーバおよびクライアントで通信量やトランスポート統計情報を定期的（約 1 秒毎）に記録します。

列名	説明
Date_Time	記録日時
Tx	送信サイズ(バイト)
Rx	受信サイズ(バイト)
AcqTx	送信帯域獲得サイズ(バイト)
AcqRx	受信帯域獲得サイズ(バイト)
Cwnd	輻輳ウィンドウサイズ(バイト)
RTT	RTT (マイクロ秒)
Retrans	再送回数
TpTx	送信スループット(bps)
TpRx	受信スループット(bps)
AvgTpTx	送信平均スループット(bps)
AvgTpRx	受信平均スループット(bps)

Date_Time は、この記録が作成された日時が記録されます。次の書式で出力されます。

yyyy/mm/dd HH:MM::SS.uuuuuu (マイクロ秒まで記載)

Tx は、この記録が作成されるまでにトランスポート層で送信されたバイト数が記録されます（差分）。

Rx は、この記録が作成されるまでにトランスポート層で受信したバイト数が記録されます（差分）。

AcqTx は、この記録が作成されるまでにトランスポート層で獲得された送信帯域のバイト数が記録されます（差分）。

AcqRx は、この記録が作成されるまでにトランスポート層で獲得された受信帯域のバイト数が記録されます（差分）。

Cwnd は、この記録が作成された時点で **TCP** の統計情報から取得した輻輳ウィンドウのサイズが記録されます。

RTT は、この記録が作成された時点で **TCP** の統計情報から取得した **RTT** が記録されます。

Retrans は、この記録が作成された時点で **TCP** の統計情報から取得した再送カウントが記録されます。

TpTx は、この記録のトランスポート層の送信スループットが記録されます。

TpRx は、この記録のトランスポート層の受信スループットが記録されます。

AvgTpTx は、この記録が作成された時点でのトランスポート層の平均送信スループットが記録されます。

AvgTpRx は、この記録が作成された時点でのトランスポート層の平均受信スループットが記録されます。

--

例：

TODO

--

PID 11168, Local 127.0.0.1:58374, Foreign 127.0.0.1:11111

Date_Time, Tx (bytes), Rx (bytes), AcqTx (bytes), AcqRx (bytes), Cwnd (bytes), RTT (usec), Retrans, TpTx (bps), TpRx (bps), AvgTpTx (bps), AvgTpRx (bps)

2018/05/25 15:05:50.142156, 0, 0, 0, 0, 218450, 17, 0, 0.000000, 0.000000, 0.000000, 0.000000

2018/05/25 15:05:52.868790, 2288, 1760, 2288, 1764, 654830, 21, 0, 6712.942871, 5163.802246, 6712.854004, 5163.731934

2018/05/25 15:05:53.321617, 4436, 2920, 4436, 2920, 654830, 12764, 0, 78369.882812, 51587.031250, 16918.314453, 11775.388672

--

3.3 システム統計

サーバ(hcpd デーモン)実行中に、通信量およびアクセス状況及び状態を定期的（約 10 秒毎）に記録します。

列名	説明
Date_Time	記録日時
Tx	送信サイズ(バイト)
Rx	受信サイズ(バイト)
AcqTx	送信帯域獲得サイズ(バイト)
AcqRx	受信帯域獲得サイズ(バイト)
Conn	着信接続数
TCPConn	着信 TCP 接続数
HpFPConn	着信 HpFP 接続数
UserConn	着信ユーザ数
WorkConn	動作中接続数
WorkTCPConn	動作中 TCP 接続数
WorkHpFPConn	動作中 HpFP 接続数

Date_Time は、この記録が作成された日時が記録されます。次の書式で出力されます。

yyyy/mm/dd HH:MM::SS.uuuuuuu （マイクロ秒まで記載）

Tx は、この記録が作成されるまでにトランスポート層で送信されたバイト数が記録されます（差分）。

Rx は、この記録が作成されるまでにトランスポート層で受信したバイト数が記録されます（差分）。

AcqTx は、この記録が作成されるまでにトランスポート層で獲得された送信帯域のバイト数が記録されます（差分）。

AcqRx は、この記録が作成されるまでにトランスポート層で獲得された受信帯域のバイト数が記録されます（差分）。

Conn は、この記録が作成されるまでに着信した接続数が記録されます（差分）。

TCPConn は、この記録が作成されるまでに着信した TCP 接続数が記録されます（差分）。

HpFPConn は、この記録が作成されるまでに着信した HpFP 接続数が記録されます（差分）。

UserConn は、この記録が作成されるまでに着信したユーザ数が記録されます（差分）。

WorkConn は、この記録が作成される時点で動作中の接続数が記録されます（現在値）。

WorkTCPConn は、この記録が作成される時点で動作中の TCP 接続数が記録されます（現在値）。

WorkHpFPConn は、この記録が作成される時点で動作中の HpFP 接続数が記録されます（現在値）。

--

例：

--

Date_Time, Tx (bytes), Rx (bytes), AcqTx (bytes), AcqRx (bytes), Conn, TCPConn, HpFPConn, UserConn, WorkConn, WorkTCPConn, WorkHpFPConn

2018/07/05 18:17:40.657653, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

2018/07/05 18:17:50.666458, 4624, 5688, 4624, 5696, 2, 1, 1, 1, 1, 0, 1

2018/07/05 18:17:52.910125, 2624, 4120, 2624, 4124, 0, 0, 0, 1, 0, 0, 0

--

4 エラーコードリファレンス

4.1 コマンド終了ステータス

各コマンドの実行時に、0 から 255 の範囲で下表に記載の終了コードを実行結果として返します。

終了コード	名前	説明
0	EXIT_SUCCESS	成功
1	EXIT_FAILURE	失敗(汎用)
21	EXIT_NOTICE	注意付き終了
40	EXIT_PENDING	保留(予約)

50	EXIT_WARN	警告(予約)
71	EXIT_ERROR_CONNECT_FAILED	接続失敗
72	EXIT_ERROR_AUTH_FAILED	認証失敗
73	EXIT_ERROR_CONNECTION_FAILURE	通信エラー
79	EXIT_ERROR_NEG_FAILED	ネゴシエーションエラー
81	EXIT_ERROR_IO_FAILURE	アプリケーション I/O エラー (ディスク、ファイルアクセス 等起因)
90	EXIT_ERROR_APP_FAILURE	アプリケーション処理エラー
91	EXIT_ERROR_APP_SETUP_FAILED	アプリケーション起動エラー
99	EXIT_ERROR_APP_ABORTED	アプリケーション中断
181	EXIT_ERROR_LOAD_ARG	コマンド引数エラー
182	EXIT_ERROR_LOAD_CONF	コマンド設定エラー
189	EXIT_ERROR_CLI_SETUP_FAILED	CLI 起動エラー
191	EXIT_ERROR_SERVICE_SETUP_FAILED	サービス起動エラー
200	EXIT_FATAL	障害(汎用)
201	EXIT_FATAL_RUNTIME_SETUP_FAILURE	実行環境起動エラー
202	EXIT_FATAL_MUTEX_SETUP_FAILURE	実行プログラム同期エラー

4.2 設定ファイルエラーコード

設定ファイルに記述エラーを検出した場合、コマンドは終了ステータス (EXIT_ERROR_LOAD_CONF) を実行結果として返します。その際、ユーザインタフェース（シェルの場合は標準エラー出力）には、検出したエラーに関するメッセージと原因の種別を表すエラーコード（下記表に記載）を表示します。エラーコードは、0 から 255 の範囲(8bit)で表されます。

このエラーは、下記の書式で表示されます。

書式 <config_path>: [<error_code>] <message>

config_path は、エラーが発生した設定ファイルのパスを表します。

error_code は、発生したエラーの種別を表すエラーコードを表します。

message は、発生したエラーメッセージを表します。

--

例:

```
/etc/hcp/hcp.conf: [052] Configuration file parse error in line 3 : PPubk  
eyAuthentication yes
```

--

エラー コード	名前	説明
0	CONF_ERROR_OK	成功(通常表示されない)
50	CONF_ERROR_NULL_NAME	設定項目名の取得に失敗
51	CONF_ERROR_EMPTY_NAME	設定項目名が空文字
52	CONF_ERROR_UNKNOWN_NAME	設定項目名が不明
53	CONF_ERROR_INVALID_NAME	設定項目名が不正
54	CONF_ERROR_INVALID_VALUE	設定項目の値が不正
60	CONF_ERROR_QUOTE_NOT_CLOSED	設定値のクオートが閉じられていない
61	CONF_ERROR_QUOTE_IN_QUOTE	設定値のクオート中にクオートが発見された
70	CONF_ERROR_NO_VALUES	設定値がなかった
71	CONF_ERROR_TOO_MANY_VALUES	設定項目の値の数が多すぎる
79	CONF_ERROR_OVER_LIMIT_NUM_OF_VALUES	設定可能な値の数の上限を超えた
80	CONF_ERROR_UNEXPECTED_NESTED_STRUCT	ネストされた構造化設定項目が検出された
81	CONF_ERROR_UNEXPECTED_ENDING_STRUCT	予期しない構造化設定項目の終了が検出された
82	CONF_ERROR_UNEXPECTED_VALUE_OF_STRUCT	予期しない構造化設定項目内の設定項目が検出された
84	CONF_ERROR_ENDING_STRUCT_NOT_FOUND	構造化設定項目の終了が見つからなかった

85	CONF_ERROR_OVER_LIMIT_NUM_OF_STRUCTS	構造化設定項目の設定可能な上限の数を超えた
99	CONF_ERROR_LINE_LEN_EXCEEDED	設定行の上限文字数を超えた
200	CONF_ERROR_CONF_NOT_FOUND	設定ファイルが見つからなかった
201	CONF_ERROR_FILE_OPEN_FAILED	設定ファイルを開くのに失敗した
255	CONF_ERROR_FAILURE	失敗

4.3 ファイル処理理由コード

`verbose` オプションを指定した場合、ファイル転送などの処理結果が標準出力に下記表記載の理由コード(Reason code)と伴に出力される場合があります。理由コードは、0 から 65536 の範囲(16bit)で 16 進数で表示されます。

理由コード(16進数)	名称	説明
0000	NO_ERROR	成功
0001	ALREADY_DONE	実行済み
2C01	NOT_MODIFIED	ファイル未更新
2C02	NOT_DIFFERENT	ファイル差分なし
2C03	NOT_PERMITTED	処理不許可
2C04	NOT_CONFIRMED_OVERWRITE	上書き許可されなかった
2C05	NOT_CREATED	作成されなかった
2C06	MODIFIED	TODO 更新済み
2C07	DIFFERENT	TODO ファイル差分あり
2C08	MODIFIED_SIZE_IDENT	更新日時ファイルサイズ同一
2C09	NOT_EXISTING	ファイルは存在していない
2C0A	EXISTING	ファイルは存在している
2C0B	NOT_MATCH_COND	処理条件不一致
2C0C	DELETED	削除された
2C0D	NOT_DELETED	削除されなかった

2C0E	NO_CONTENT_OPERATED	コンテンツ未処理
2C0F	EXCLUSION_MATCH	除外条件一致
2FFD	DELETE_EXCLUDED	除外ファイル削除
2FFE	APP_MAX_EXCEEDED	アプリ指定上限超過
2FFF	APP_SKIP	アプリ指定条件でスキップ
9001	UNKNOWN_ENCODING	不明な文字エンコーディング方式
9002	UNKNOWN_BCIPH	不明な暗号方式
9003	UNKNOWN_COMPR	不明な圧縮方式
9004	AUTH_REQUIRED	認証が必要
9005	FILE_ENTRY_NOT_FOUND	ファイル情報が見つからなかった
9006	FILE_ENTRY_ERROR	ファイル情報エラー
9007	FIND_FILE_ERROR	ファイル探索エラー
9008	AUTH_FAILED	認証失敗
9009	NEG_FAILED	ネゴシエーション失敗
900A	KEY_EXHG_REQUIRED	セキュリティ鍵の交換が必要
900B	APP_NOT_SUPPORTED	アプリ（コマンド）非対応
9FFB	SESS_IDLE_TIMEOUT	セッションアイドルタイムアウト
9FFC	UNEXP_TERM	予期しない終了
9FFD	APP_ABORT	アプリによる中断
9FFE	SESS_ABORT	セッションによる中断
9FFF	FILE_REQQ_FULL	ファイルキュー不足
A001	FILE_NOT_FOUND	ファイルが見つからなかった
A002	FILE_ALREADY_EXISTS	ファイルが既に存在する
A003	FILE_ACCESS_DENIED	ファイルアクセス不許可
A004	DIGEST_ERROR	データ完全性検査エラー
A005	WRITE_ERROR	ファイル書込みエラー
A006	CREATE_DIR_ERROR	ディレクトリ作成失敗
A007	CREATE_LINK_ERROR	リンク作成失敗

A008	READ_ERROR	ファイル読み込みエラー
A009	IS_DIR	ディレクトリ検出
A00A	IS_NOT_DIR	ディレクトリ非検出
A00B	RENAME_ERROR	ファイル名変更エラー
A00C	DELETE_ERROR	ファイル削除エラー
A00D	FILE_IDENT	ファイルが同一
A00E	OP_NOT_PERMITTED	操作不許可
A00F	MODIFY_ERROR	ファイル情報更新エラー
A010	CANNOT_OVERWRITE_DIR	ディレクトリ上書き不可
A011	CANNOT_OVERWRITE_NON_DIR	非ディレクトリ上書き不可
A012	DEST_IS_NOT_DIR	宛先がディレクトリでない
A013	OMIT_DIR	ディレクトリ省略
A014	TMP_FILE_SETUP_ERROR	一時ファイル生成エラー
A015	TMP_FILE_FIX_ERROR	一時ファイル確定処理エラー
AFFD	MISS_LAST_MODIFIED	ファイル更新確認要求欠損
AFFE	FILE_OUT_OF_DOC_ROOT	ドキュメントルート外アクセス
AFFF	FILE_SIZE_OVER_LIMIT	ファイルサイズ上限を超えた
B001	BUSY	サーバビジー
B002	CONNECT_FAILED	接続失敗
B003	REFUSED	サーバ拒否
BFF3	SESS_ALREADY_SHUTDOWN	セッションシャットダウン済み
BFF4	SESS_COLLISION	セッション競合
BFF5	SCH_BIND_FAILED	セカンダリ接続バインド失敗
BFF6	SESS_ID_NOT_FOUND	セッション識別子不明
BFF8	INCOMPATIBLE_FEATURE	機能非互換
BFF9	RL_ABORT	リソース制限による中断
BFFA	INCOMPATIBLE_PROTO_VERSION	プロトコル非互換
BFFB	SETUP_FAILURE	セットアップ失敗

BFFC	NET_UNAVAIL	通信非有効状態
BFFD	DISK_FULL	ディスク不足
BFFE	SESS_NOT_FOUND	セッションが見つからなかった
BFFF	PROTO_ERROR	プロトコルエラー
FFFD	GC_ABORT	GC（プロセス停止時など）による中断
FFFE	INTERNAL_ERROR	内部エラー
FFFF	GENERAL_ERROR	失敗

4.4 サービスシグナルハンドリング

サービス（hcpd デーモン）は、シグナルを使用して後述の様な制御を行うことができます。

次のシグナルをサービスの停止シグナルとしてキャッチして処理します。

シグナル名称	説明
SIGINT	ターミナル割込み
SIGTERM	ターミナル終了

SIGINT は、通信中の処理の終了を待機してから停止します。SIGTERM は、通信中の処理を中断して停止します。

次のシグナルをサービスの再起動（設定の再読み込み）シグナルとしてキャッチして処理します。

シグナル名称	説明
SIGHUP	ハングアップ
SIGUSR1	USR1 シグナル

次のシグナルは無視します。

シグナル名称	説明
SIGALRM	アラーム
SIGPIPE	パイプ書き込みエラー

これら以外のシグナルは処理しません（オペレーティングシステムが規定する動作を行います）。

シグナルの番号（数値）はプラットフォーム毎に異なることがあります。各オペレーティングシステムのマニュアルをご参照ください。

4.5 アプリケーションシグナルハンドリング

クライアントのコマンド（hcp など）やサービスを特権分離で使用している場合に生成される子プロセスで実行されるアプリケーション処理は、シグナルを使用して後述の様な制御を行うことができます。

次のシグナルをアプリケーション中断シグナルとしてキャッチして処理します。実行中のアプリケーション処理を中断し、リソースを回収してプロセスを停止します。

シグナル名称	説明	Windows
SIGINT	ターミナル割込み	○
SIGTERM	ターミナル終了	○
SIGUSR1	USR1 シグナル	×

次のシグナルをリソース制限に基づくアプリケーション中断シグナルとしてキャッチして処理します。実行中のアプリケーション処理を中断し、リソースを回収してプロセスを停止します。

シグナル名称	説明	Windows
SIGUSR2	USR2 シグナル	×
SIGXCPU	CPU 制限	×
SIGXFSZ	ファイルサイズ制限	×

次のシグナルをリソース回収シグナルとしてキャッチして処理します。可能な限りリソース（HpFP ソケットディスクリプタなど）を解放し、シグナルを再送信（re-raise）してプロセスを即時停止します。

シグナル名称	説明	Windows
SIGHUP	ハングアップ	×

このシグナルは、サービスの子プロセスに対しては使用しないでください。サービス全体の継続稼働に支障が生じることがあります（サービスの強制停止及び再起動が必要になる場合があります）。

次のシグナルは無視します。

シグナル名称	説明	Windows
SIGALRM	アラーム	×
SIGPIPE	パイプ書込みエラー	×

これら以外のシグナル（SIGKILL、SIGABORT など）は処理しません（オペレーティングシステムが規定する動作を行います）。一部のリソース（HpFP ソケットデイスクリプタなど）は使用できなくなります。

これらのシグナルは、サービスの子プロセスに対しては使用しないでください。サービス全体の継続稼働に支障が生じることがあります（サービスの強制停止及び再起動が必要になる場合があります）。

シグナルの番号（数値）はプラットフォーム毎に異なることがあります。各オペレーティングシステムのマニュアルをご参照ください。

5 コマンド実行モード

HCP tools では、クライアントのコマンド実行モードを次の二つを提供します。

- 単一起動モード (Single running mode)
- 多重起動モード (Multiple running mode)

多重起動モードは、コマンドの `multi-run` オプションを指定すると動作します。本オプションを指定しない場合は、単一起動モードとなります。多重起動モードは、作業ディレクトリを複数のユーザで共有する場合は、バックグラウンド実行を行う場合などに利用します。

以下、それぞれのモードでの実行方法、注意事項などを説明します。

5.1 単一起動モード

本モードは、特定の作業ディレクトリで単一のユーザがシーケンシャルにコマンドを実行する様な利用形態で使します。コマンドの終了ステータスは、通常のコマンドと同様の方法で取得します。

```
--
例：
(Linux)
[user@localhost ~]$ hcp src.txt dest.txt
...
```

```
[user@localhost ~]$ echo $?
0
(Windows)
C:\Users\user> hcp src.txt dest.txt
...
C:\Users\user> echo %ERRORLEVEL%
0
--
```

コマンドの実行記録は、-v オプションで標準出力に出力しない場合は、カレントディレクトリに次の名前で実行毎に初期化（トランケート）が行われて記録されます。

```
.hcp.out (Linux)
_hcp.out (Windows)
```

アプリケーションのログは、デフォルトでは標準出力に下記の様に出力されますが、-l オプションでファイルパスを指定すると、そのファイルに出力されます。

```
2019/11/01 13:33:35 00007fae521a3b80:INFO :Set bandwidth outbound to unlimited since the specified value is over 5Gbps (given limit : 10000000000 bps).
2019/11/01 13:33:35 00007fae521a3b80:INFO :Set bandwidth inbound to unlimited since the specified value is over 5Gbps (given limit : 10000000000 bps).
2019/11/01 13:33:38 00007fae4aa4a700:INFO :A transport timeout monitor was setup (sock=5, _recv). Monitoring the timeout ...
2019/11/01 13:33:38 00007fae4aa4a700:INFO :A transport timeout monitor becomes disabled (sock=5, ret=0, errno=0, _recv).
```

統計記録（アプリケーション統計、トランスポート統計）は、保存先を指定しない場合はカレントディレクトリに次の名前で保存されます。これらの記録は、トランケートされず追記されます。

```
.hcp.statistics.application (Linux)
.hcp.statistics.transport.tcp (Linux)
.hcp.statistics.transport.hpfp (Linux)
_hcp.statistics.application (Windows)
_hcp.statistics.transport.tcp (Windows)
_hcp.statistics.transport.hpfp (Windows)
```

統計記録は、-L オプションを使用すると、出力先のディレクトリとファイル名のプレフィックス（接頭辞）を指定して出力できます。

```
--
例 :
[root@localhost ~]# hcp -L /var/tmp/.hcp.statistics2 ...
```

```
// 出力先ディレクトリ : /var/tmp
// ファイル名プレフィックス : .hcp.statistics2
--
```

5.2 多重起動モード

本モードは、単一のディレクトリを複数のユーザで共有して作業を行う場合や、コマンドをバックグラウンドで起動して単一のユーザで複数コマンドを同時に実行する場合などに利用します。コマンドの終了ステータスは、単一起動モードと同様の方法で取得するか、結果出力に記録される EXIT パラメータなどを参照します。

```
--
例 :
[user@localhost ~]$ hcp --multi-run=/var/tmp ...
2019/11/01 14:02:06 00007f35da531b80:INFO :A configuration file (/home/user/.hcp/hcp.conf) was not found. So it was skipped.
2019/11/01 14:02:06 00007f35da531b80:      :Recording on multi-running is configured (ID=hcp.mr.20191101_140206_442.15279, dir=/var/tmp, out=ATR).
2019/11/01 14:02:06 00007f35da531b80:INFO :Set bandwidth outbound to unlimited since the specified value is over 5Gbps (given limit : 10000000000 bps).
2019/11/01 14:02:06 00007f35da531b80:INFO :Set bandwidth inbound to unlimited since the specified value is over 5Gbps (given limit : 10000000000 bps).
Login as:user
Password:
2019/11/01 14:02:09 00007f35d2dd8700:INFO :A transport timeout monitor was setup (sock=5, _recv). Monitoring the timeout ...
2019/11/01 14:02:09 00007f35d2dd8700:INFO :A transport timeout monitor becomes disabled (sock=5, ret=0, errno=0, _recv).

[user@localhost ~]$ hcp -l hcp.log --multi-run=/var/tmp ...
2019/11/01 14:04:23 00007fb04eac3b80:INFO :A configuration file (/home/user/.hcp/hcp.conf) was not found. So it was skipped.
2019/11/01 14:04:23 00007fb04eac3b80:      :Logging on multi-running is configured (ID=hcp.mr.20191101_140423_458.15318, dir=/var/tmp, path=/var/tmp/hcp.mr.20191101_140423_458.15318.log).
Login as:user
Password:
[user@localhost ~]$ less /var/tmp/hcp.mr.20191101_140423_458.15318.log
2019/11/01 14:04:23 00007fb04eac3b80:      :Recording on multi-running is configured (ID=hcp.mr.20191101_140423_458.15318, dir=/var/tmp, out=ATR).
2019/11/01 14:04:23 00007fb04eac3b80:INFO :Set bandwidth outbound to unlimited since the specified value is over 5Gbps (given limit : 10000000000 bps).
```

```
2019/11/01 14:04:23 00007fb04eac3b80:INFO :Set bandwidth inbound to unlimited since the specified value is over 5Gbps (given limit : 10000000000 bps).
```

```
2019/11/01 14:04:26 00007fb04736a700:INFO :A transport timeout monitor was setup (sock=6, _recv). Monitoring the timeout ...
```

```
[user@localhost ~]$ cat /var/tmp/hcp.mr.20191101_140423_458.15318.out | grep -e EXIT
EXIT 0 REASON 0000
--
```

本モードでは、コマンド実行開始時に—multi-run オプションに指定されたディレクトリに次の様なロックファイルを作成します。

hcp.mr.<YYYYMMDD_hhmmss_msec>.<pid>.lock

コマンド毎にこのロックファイルのプレフィックス（".lock"より前の文字列）に文字列を付加したファイル名で次の記録を保存します。

- アプリケーション統計
- トランスポート統計
- 結果出力
- アプリケーションログ

また、本モードでは、次のエラーステータスで終了する場合は、これらの記録は提供されません。単一起動モードと同様にコマンドラインから終了ステータスの確認を行います。

- EXIT_ERROR_LOAD_ARG (181)
- EXIT_ERROR_LOAD_CONF (182)
- EXIT_ERROR_CLI_SETUP_FAILED (189)

バックグラウンドモードを使用する場合は、上記ケースで停止した場合終了ステータス等の確認が困難となりますので、事前に設定構成を担保するなどご注意ください。

コマンドの実行記録は、—multi-run オプションに従ってファイルに出力される場合は、次のファイル名で記録されます。

hcp.mr.<YYYYMMDD_hhmmss_msec>.<pid>.out

コマンドの-v オプションが適用される場合は、ファイルに保存されず標準出力に出力されます。

アプリケーションのログは、`—multi-run` オプションに従ってファイルに出力される場合は、次のファイル名で記録されます。

```
hcp.mr.<YYYYMMDD_hhmmss_msec>.<pid>.log
```

統計記録（アプリケーション統計、トランスポート統計）は、`—multi-run` オプションに従ってファイルに出力される場合は、次のファイル名で記録されます。

```
hcp.mr.<YYYYMMDD_hhmmss_msec>.<pid>.statistics.application
hcp.mr.<YYYYMMDD_hhmmss_msec>.<pid>.statistics.transport.tcp
hcp.mr.<YYYYMMDD_hhmmss_msec>.<pid>.statistics.transport.hpfp
```

6 HCP tools 制約事項一覧

名称	制約
設定項目の値の数	8 個まで
設定項目の行文字数	256
設定項目のエラーメッセージ長	1024
DocPoint 定義数	10 個まで
users の行文字数	256
passwd の行文字数	256
ファイルサイズ	8EiB 符号付き倍長整数の最大値
ファイルパス長	2048 文字
ファイルパスに使用できない文字	非印字文字、OS 非対応文字、改行、など
ファイルキャッシュエントリ数	100000 個
処理待機中ファイルの結果出力数	1000 件まで

※ファイルパス長は、プラットフォームのファイルシステムごとに実際のパス長の制限が異なります。Windows では、概ねドライブレターを含め 260 文字まで使用できます。

7 表記法について

対応 OS 及び OS ごとの既定値、設定値または例などの記載において表記される名称は以下のルールに準じます。

次の個々の OS とバージョンは括弧に記載されるラベルで表します。

- Red Hat Enterprise Linux 7 / CentOS 7 (RHEL7)
- Red Hat Enterprise Linux 8 / AlmaLinux 8 (RHEL8)
- SUSE Linux Enterprise 15 SP1 / openSUSE Leap 15.1 (SLE15.1)
- SUSE Linux Enterprise 15 SP2 / openSUSE Leap 15.2 (SLE15.2)
- Ubuntu 16.04 LTS / Debian 9 (Ubuntu16.04)
- Ubuntu 18.04 LTS / Debian 10 (Ubuntu18.04)
- Ubuntu 20.04 LTS / Debian 11 (Ubuntu20.04)
- Raspberry Pi OS 9 (Raspbian9)
- Windows 10 (Win10)
- Windows Server 2012 R2 (WinServ2012R2)
- Windows Server 2016 (WinServ2016)

また、これらの OS のグループを次の括弧に記載されるラベルで表します。

- RHEL7 / RHEL8 (RHEL)
- SLE15.1 / SLE15.2 (SLE)
- Ubuntu16.04 / Ubuntu18.04 / Ubuntu20.04 (Debian)
- Raspbian9 (Raspbian)
- Win10 / WinServ2012R2 / WinServ2016 (Windows)

また、Linux 系ディストリビューションのグループを次の括弧に記載されるラベルで表します。

- RHEL / SLE / Debian / Raspbian (Linux)
- RHEL / SLE / Debian (Linux.x86)

8 改訂履歴

編集日	変更点
2021/11/29	設定項目説明追加、hsync、hchmod、hchown の説明を追加、コマンド説明編集
2021/09/10	POSIX ユーザ・グループアクセス制限説明追加、他説明追加
2021/05/16	デフォルト値改訂反映、設定項目説明追加、アプリ統計説明追加
2021/04/23	デフォルト値の誤記訂正
2021/04/18	接続多重化、WebSocket 関連設定項目説明追加
2021/02/09	設定項目・コマンドパラメータ説明追加、出力例改訂など
2020/09/03	既定値等 Linux 明記、表記法追記、archivecheck 追記、予約表記訂正

2020/07/30	UseProperCopyAndSync 説明改訂
2020/07/21	予約表記訂正、FileOperationLog 説明改訂、シグナル説明追加
2020/05/15	設定項目説明追加、エラーコード追加
2020/04/23	MAC 無効化説明追加
2020/02/10	FileOperationLog 説明追加
2019/12/16	ホームディレクトリ検査設定項目説明追加
2019/12/11	ログローテーション設定項目説明追加、ファイルパス長注記
2019/11/12	誤記訂正、秘密鍵探索規則注記
2019/11/05	項番出力訂正
2019/11/01	誤記訂正、記載改訂、設定項目説明追加
2019/06/07	コマンド記述追記・編集、理由コード追記
2019/04/26	users サンプル記述訂正
2019/04/25	オプション説明追加、記載改訂
2019/02/01	オプション説明追加
2019/01/30	誤記訂正、Include 設定項目説明追加
2019/01/29	誤記訂正
2019/01/21	サーバ鍵ファイルパス仕様訂正
2019/01/20	追加コマンド、Win 版対応記述追加
2018/07/30	アプリ統計記述編集及び出力例改訂
2018/07/30	終了ステータス及び理由コード改訂