```
**********************************************************
   37346 Thu Jan 17 15:17:54 2013
new/usr/src/cmd/ssh/libssh/common/readconf.c
3477 SunSSH config should accept TCPKeepAlive as synonym for KeepAlive
Reviewed by: Jerry Jelinek <jerry@joyent.com>
**********************************************************
/*
 * Author: Tatu Ylonen <ylo@cs.hut.fi>
 * Copyright (c) 1995 Tatu Ylonen <ylo@cs.hut.fi>, Espoo, Finland
 *                    All rights reserved
 * Functions for reading the configuration files.
 *
 * As far as I am concerned, the code I have written for this software
 * can be used freely for any purpose.  Any derived versions of this
 * software must be clearly marked as such, and if the derived work is
 * incompatible with the protocol description in the RFC file, it must be
 * called by a name other than "ssh" or "Secure Shell".
 */
/*
 * Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
 * Use is subject to license terms.
 * Copyright 2013 Joyent, Inc.  All rights reserved.
 */

#include "includes.h"
RCSID("$OpenBSD: readconf.c,v 1.100 2002/06/19 00:27:55 deraadt Exp $");

#include "ssh.h"
#include "xmalloc.h"
#include "compat.h"
#include "cipher.h"
#include "pathnames.h"
#include "log.h"
#include "readconf.h"
#include "match.h"
#include "misc.h"
#include "kex.h"
#include "mac.h"

/* Format of the configuration file:

   # Configuration data is parsed as follows:
   #  1. command line options
   #  2. user-specific file
   #  3. system-wide file
   # Any configuration value is only changed the first time it is set.
   # Thus, host-specific definitions should be at the beginning of the
   # configuration file, and defaults at the end.

   # Host-specific declarations.  These may override anything above.  A single
   # host may match multiple declarations; these are processed in the order
   # that they are given in.

   Host *.ngs.fi ngs.fi
     User foo

   Host fake.com
     HostName another.host.name.real.org
     User blaah
     Port 34289
     ForwardX11 no
     ForwardAgent no

   Host books.com
     RemoteForward 9999 shadows.cs.hut.fi:9999
     Cipher 3des
```



```
   Host fascist.blob.com
     Port 23123
     User tylonen
     RhostsAuthentication no
     PasswordAuthentication no

   Host puukko.hut.fi
     User t35124p
     ProxyCommand ssh-proxy %h %p

   Host *.fr
     PublicKeyAuthentication no

   Host *.su
     Cipher none
     PasswordAuthentication no

   # Defaults for various options
   Host *
     ForwardAgent no
     ForwardX11 no
     RhostsAuthentication yes
     PasswordAuthentication yes
     RSAAuthentication yes
     RhostsRSAAuthentication yes
     StrictHostKeyChecking yes
     KeepAlives no
     IdentityFile ~/.ssh/identity
     Port 22
     EscapeChar ~

*/

/* Keyword tokens. */

typedef enum {
        oBadOption,
        oForwardAgent, oForwardX11, oForwardX11Trusted, oGatewayPorts,
        oRhostsAuthentication,
        oPasswordAuthentication, oRSAAuthentication,
        oChallengeResponseAuthentication, oXAuthLocation,
#if defined(KRB4) || defined(KRB5)
        oKerberosAuthentication,
#endif
#ifdef GSSAPI
        oGssKeyEx, oGssAuthentication, oGssDelegateCreds,
#ifdef GSI
        oGssGlobusDelegateLimitedCreds,
#endif /* GSI */
#endif /* GSSAPI */
#if defined(AFS) || defined(KRB5)
        oKerberosTgtPassing,
#endif
#ifdef AFS
        oAFSTokenPassing,
#endif
        oIdentityFile, oHostName, oPort, oCipher, oRemoteForward, oLocalForward,
        oUser, oHost, oEscapeChar, oRhostsRSAAuthentication, oProxyCommand,
        oGlobalKnownHostsFile, oUserKnownHostsFile, oConnectionAttempts,
        oBatchMode, oCheckHostIP, oStrictHostKeyChecking, oCompression,
        oCompressionLevel, oKeepAlives, oNumberOfPasswordPrompts,
        oUsePrivilegedPort, oLogLevel, oCiphers, oProtocol, oMacs,
        oGlobalKnownHostsFile2, oUserKnownHostsFile2, oPubkeyAuthentication,
        oKbdInteractiveAuthentication, oKbdInteractiveDevices, oHostKeyAlias,
        oDynamicForward, oPreferredAuthentications, oHostbasedAuthentication,
```

```
 127         oHostKeyAlgorithms, oBindAddress, oSmartcardDevice,
 128         oClearAllForwardings, oNoHostAuthenticationForLocalhost,
 129         oFallBackToRsh, oUseRsh, oConnectTimeout, oHashKnownHosts,
 130         oServerAliveInterval, oServerAliveCountMax, oDisableBanner,
 131         oIgnoreIfUnknown, oRekeyLimit, oUseOpenSSLEngine,
 132         oDeprecated
 133 } OpCodes;
_____unchanged_portion_omitted_
 141         { "forwardagent", oForwardAgent },
 142         { "forwardx11", oForwardX11 },
 143         { "forwardx11trusted", oForwardX11Trusted },
 144         { "xauthlocation", oXAuthLocation },
 145         { "gatewayports", oGatewayPorts },
 146         { "useprivilegedport", oUsePrivilegedPort },
 147         { "rhostsauthentication", oRhostsAuthentication },
 148         { "passwordauthentication", oPasswordAuthentication },
 149         { "kbdinteractiveauthentication", oKbdInteractiveAuthentication },
 150         { "kbdinteractivedevices", oKbdInteractiveDevices },
 151         { "rsaauthentication", oRSAAuthentication },
 152         { "pubkeyauthentication", oPubkeyAuthentication },
 153         { "dsaauthentication", oPubkeyAuthentication },           /* alias */
 154         { "rhostsrsaauthentication", oRhostsRSAAuthentication },
 155         { "hostbasedauthentication", oHostbasedAuthentication },
 156         { "challengeresponseauthentication", oChallengeResponseAuthentication },
 157         { "skeyauthentication", oChallengeResponseAuthentication }, /* alias */
 158         { "tisauthentication", oChallengeResponseAuthentication },  /* alias */
 159 #if defined(KRB4) || defined(KRB5)
 160         { "kerberosauthentication", oKerberosAuthentication },
 161 #endif
 162 #ifdef GSSAPI
 163         { "gssapikeyexchange", oGssKeyEx },
 164         { "gssapiauthentication", oGssAuthentication },
 165         { "gssapidelegatecredentials", oGssDelegateCreds },
 166         { "gsskeyex", oGssKeyEx },                          /* alias */
 167         { "gssauthentication", oGssAuthentication },        /* alias */
 168         { "gssdelegatecreds", oGssDelegateCreds },          /* alias */
 169 #ifdef GSI
 170         /* For backwards compatability with old 1.2.27 client code */
 171         { "forwardgssapiglobusproxy", oGssDelegateCreds }, /* alias */
 172         { "forwardgssapiglobuslimitedproxy", oGssGlobusDelegateLimitedCreds },
 173 #endif /* GSI */
 174 #endif /* GSSAPI */
 175 #if defined(AFS) || defined(KRB5)
 176         { "kerberostgtpassing", oKerberosTgtPassing },
 177 #endif
 178 #ifdef AFS
 179         { "afstokenpassing", oAFSTokenPassing },
 180 #endif
 181         { "fallbacktorsh", oFallBackToRsh },
 182         { "usersh", oUseRsh },
 183         { "identityfile", oIdentityFile },
 184         { "identityfile2", oIdentityFile },                 /* alias */
 185         { "hostname", oHostName },
 186         { "hostkeyalias", oHostKeyAlias },
 187         { "proxycommand", oProxyCommand },
 188         { "port", oPort },
 189         { "cipher", oCipher },
 190         { "ciphers", oCiphers },
 191         { "macs", oMacs },
 192         { "protocol", oProtocol },
 193         { "remoteforward", oRemoteForward },
 194         { "localforward", oLocalForward },
 195         { "user", oUser },
 196         { "host", oHost },
 197         { "escapechar", oEscapeChar },
 198         { "globalknownhostsfile", oGlobalKnownHostsFile },
```



```
 199         { "userknownhostsfile", oUserKnownHostsFile },          /* obsolete */
 200         { "globalknownhostsfile2", oGlobalKnownHostsFile2 },
 201         { "userknownhostsfile2", oUserKnownHostsFile2 },        /* obsolete */
 202         { "connectionattempts", oConnectionAttempts },
 203         { "batchmode", oBatchMode },
 204         { "checkhostip", oCheckHostIP },
 205         { "stricthostkeychecking", oStrictHostKeyChecking },
 206         { "compression", oCompression },
 207         { "compressionlevel", oCompressionLevel },
 208         { "tcpkeepalive", oKeepAlives },
 209         { "keepalive", oKeepAlives },                           /* obsolete */
 207         { "keepalive", oKeepAlives },
 210         { "numberofpasswordprompts", oNumberOfPasswordPrompts },
 211         { "loglevel", oLogLevel },
 212         { "dynamicforward", oDynamicForward },
 213         { "preferredauthentications", oPreferredAuthentications },
 214         { "hostkeyalgorithms", oHostKeyAlgorithms },
 215         { "bindaddress", oBindAddress },
 216         { "smartcarddevice", oSmartcardDevice },
 217         { "clearallforwardings", oClearAllForwardings },
 218         { "nohostauthenticationforlocalhost", oNoHostAuthenticationForLocalhost
 219         { "rekeylimit", oRekeyLimit },
 220         { "connecttimeout", oConnectTimeout },
 221         { "serveraliveinterval", oServerAliveInterval },
 222         { "serveralivecountmax", oServerAliveCountMax },
 223         { "disablebanner", oDisableBanner },
 224         { "hashknownhosts", oHashKnownHosts },
 225         { "ignoreifunknown", oIgnoreIfUnknown },
 226         { "useopensslengine", oUseOpenSSLEngine },
 227         { NULL, oBadOption }
 228 };
_____unchanged_portion_omitted_
```

```
**********************************************************
   45024 Thu Jan 17 15:17:54 2013
new/usr/src/cmd/ssh/sshd/servconf.c
3477 SunSSH config should accept TCPKeepAlive as synonym for KeepAlive
Reviewed by: Jerry Jelinek <jerry@joyent.com>
**********************************************************
```

```
/*
 * Copyright (c) 1995 Tatu Ylonen <ylo@cs.hut.fi>, Espoo, Finland
 *                    All rights reserved
 *
 * As far as I am concerned, the code I have written for this software
 * can be used freely for any purpose.  Any derived versions of this
 * software must be clearly marked as such, and if the derived work is
 * incompatible with the protocol description in the RFC file, it must be
 * called by a name other than "ssh" or "Secure Shell".
 */
/*
 * Copyright (c) 2001, 2010, Oracle and/or its affiliates. All rights reserved.
 * Copyright 2013 Joyent, Inc.  All rights reserved.
 */

#include "includes.h"
RCSID("$OpenBSD: servconf.c,v 1.115 2002/09/04 18:52:42 stevesk Exp $");

#ifdef HAVE_DEFOPEN
#include <deflt.h>
#endif /* HAVE_DEFOPEN */

#if defined(KRB4)
#include <krb.h>
#endif
#if defined(KRB5)
#ifdef HEIMDAL
#include <krb.h>
#else
/* Bodge - but then, so is using the kerberos IV KEYFILE to get a Kerberos V
 * keytab */
#define KEYFILE "/etc/krb5.keytab"
#endif
#endif
#ifdef AFS
#include <kafs.h>
#endif

#include "ssh.h"
#include "log.h"
#include "buffer.h"
#include "servconf.h"
#include "xmalloc.h"
#include "compat.h"
#include "pathnames.h"
#include "tildexpand.h"
#include "misc.h"
#include "cipher.h"
#include "kex.h"
#include "mac.h"
#include "auth.h"
#include "match.h"
#include "groupaccess.h"

static void add_listen_addr(ServerOptions *, char *, u_short);
static void add_one_listen_addr(ServerOptions *, char *, u_short);

extern Buffer cfg;

/* AF_UNSPEC or AF_INET or AF_INET6 */
```



```
extern int IPv4or6;

/*
 * Initializes the server options to their initial (unset) values. Some of those
 * that stay unset after the command line options and configuration files are
 * read are set to their default values in fill_default_server_options().
 */
void
initialize_server_options(ServerOptions *options)
{
        (void) memset(options, 0, sizeof(*options));

        /* Standard Options */
        options->num_ports = 0;
        options->ports_from_cmdline = 0;
        options->listen_addrs = NULL;
        options->num_host_key_files = 0;
        options->pid_file = NULL;
        options->server_key_bits = -1;
        options->login_grace_time = -1;
        options->key_regeneration_time = -1;
        options->permit_root_login = PERMIT_NOT_SET;
        options->ignore_rhosts = -1;
        options->ignore_user_known_hosts = -1;
        options->print_motd = -1;
        options->print_lastlog = -1;
        options->x11_forwarding = -1;
        options->x11_display_offset = -1;
        options->x11_use_localhost = -1;
        options->xauth_location = NULL;
        options->strict_modes = -1;
        options->keepalives = -1;
        options->log_facility = SYSLOG_FACILITY_NOT_SET;
        options->log_level = SYSLOG_LEVEL_NOT_SET;
        options->rhosts_authentication = -1;
        options->rhosts_rsa_authentication = -1;
        options->hostbased_authentication = -1;
        options->hostbased_uses_name_from_packet_only = -1;
        options->rsa_authentication = -1;
        options->pubkey_authentication = -1;
#ifdef GSSAPI
        options->gss_authentication = -1;
        options->gss_keyex = -1;
        options->gss_store_creds = -1;
        options->gss_use_session_ccache = -1;
        options->gss_cleanup_creds = -1;
#endif
#if defined(KRB4) || defined(KRB5)
        options->kerberos_authentication = -1;
        options->kerberos_or_local_passwd = -1;
        options->kerberos_ticket_cleanup = -1;
#endif
#if defined(AFS) || defined(KRB5)
        options->kerberos_tgt_passing = -1;
#endif
#ifdef AFS
        options->afs_token_passing = -1;
#endif
        options->password_authentication = -1;
        options->kbd_interactive_authentication = -1;
        options->challenge_response_authentication = -1;
        options->pam_authentication_via_kbd_int = -1;
        options->permit_empty_passwd = -1;
        options->permit_user_env = -1;
        options->compression = -1;
        options->allow_tcp_forwarding = -1;
```

```
127         options->num_allow_users = 0;
128         options->num_deny_users = 0;
129         options->num_allow_groups = 0;
130         options->num_deny_groups = 0;
131         options->ciphers = NULL;
132         options->macs = NULL;
133         options->protocol = SSH_PROTO_UNKNOWN;
134         options->gateway_ports = -1;
135         options->num_subsystems = 0;
136         options->max_startups_begin = -1;
137         options->max_startups_rate = -1;
138         options->max_startups = -1;
139         options->banner = NULL;
140         options->verify_reverse_mapping = -1;
141         options->client_alive_interval = -1;
142         options->client_alive_count_max = -1;
143         options->authorized_keys_file = NULL;
144         options->authorized_keys_file2 = NULL;

146         options->max_auth_tries = -1;
147         options->max_auth_tries_log = -1;

149         options->max_init_auth_tries = -1;
150         options->max_init_auth_tries_log = -1;

152         options->lookup_client_hostnames = -1;
153         options->use_openssl_engine = -1;
154         options->chroot_directory = NULL;
155         options->pre_userauth_hook = NULL;
156         options->pam_service_name = NULL;
157         options->pam_service_prefix = NULL;
158 }
______unchanged_portion_omitted_
442         /* Portable-specific options */
443         { "PAMAuthenticationViaKbdInt", sPAMAuthenticationViaKbdInt, SSHCFG_GLOB
444         /* Standard Options */
445         { "port", sPort, SSHCFG_GLOBAL },
446         { "hostkey", sHostKeyFile, SSHCFG_GLOBAL },
447         { "hostdsakey", sHostKeyFile, SSHCFG_GLOBAL },                  /* alias
448         { "pidfile", sPidFile, SSHCFG_GLOBAL },
449         { "serverkeybits", sServerKeyBits, SSHCFG_GLOBAL },
450         { "logingracetime", sLoginGraceTime, SSHCFG_GLOBAL },
451         { "keyregenerationinterval", sKeyRegenerationTime, SSHCFG_GLOBAL },
452         { "permitrootlogin", sPermitRootLogin, SSHCFG_ALL },
453         { "syslogfacility", sLogFacility, SSHCFG_GLOBAL },
454         { "loglevel", sLogLevel, SSHCFG_GLOBAL },
455         { "rhostsauthentication", sRhostsAuthentication, SSHCFG_GLOBAL },
456         { "rhostsrsaauthentication", sRhostsRSAAuthentication, SSHCFG_ALL },
457         { "hostbasedauthentication", sHostbasedAuthentication, SSHCFG_ALL },
458         { "hostbasedusesnamefrompacketonly", sHostbasedUsesNameFromPacketOnly },
459         { "rsaauthentication", sRSAAuthentication, SSHCFG_ALL },
460         { "pubkeyauthentication", sPubkeyAuthentication, SSHCFG_ALL },
461         { "dsaauthentication", sPubkeyAuthentication, SSHCFG_GLOBAL },  /* alias
462 #ifdef GSSAPI
463         { "gssapiauthentication", sGssAuthentication, SSHCFG_ALL },
464         { "gssapikeyexchange", sGssKeyEx,   SSHCFG_GLOBAL },
465         { "gssapistoredelegatedcredentials", sGssStoreDelegCreds, SSHCFG_GLOBAL
466         { "gssauthentication", sGssAuthentication, SSHCFG_GLOBAL },     /* alias
467         { "gsskeyex", sGssKeyEx, SSHCFG_GLOBAL },       /* alias */
468         { "gssstoredelegcreds", sGssStoreDelegCreds, SSHCFG_GLOBAL },   /* alias
469 #ifndef SUNW_GSSAPI
470         { "gssusesessionccache", sGssUseSessionCredCache, SSHCFG_GLOBAL },
471         { "gssusesessioncredcache", sGssUseSessionCredCache, SSHCFG_GLOBAL },
472         { "gsscleanupcreds", sGssCleanupCreds, SSHCFG_GLOBAL },
473 #endif /* SUNW_GSSAPI */
474 #endif
```



```
475 #if defined(KRB4) || defined(KRB5)
476         { "kerberosauthentication", sKerberosAuthentication, SSHCFG_ALL },
477         { "kerberosorlocalpasswd", sKerberosOrLocalPasswd, SSHCFG_GLOBAL },
478         { "kerberosticketcleanup", sKerberosTicketCleanup, SSHCFG_GLOBAL },
479 #endif
480 #if defined(AFS) || defined(KRB5)
481         { "kerberostgtpassing", sKerberosTgtPassing, SSHCFG_GLOBAL },
482 #endif
483 #ifdef AFS
484         { "afstokenpassing", sAFSTokenPassing, SSHCFG_GLOBAL },
485 #endif
486         { "passwordauthentication", sPasswordAuthentication, SSHCFG_ALL },
487         { "kbdinteractiveauthentication", sKbdInteractiveAuthentication, SSHCFG_
488         { "challengeresponseauthentication", sChallengeResponseAuthentication, S
489         { "skeyauthentication", sChallengeResponseAuthentication, SSHCFG_GLOBAL
490         { "checkmail", sDeprecated, SSHCFG_GLOBAL },
491         { "listenaddress", sListenAddress, SSHCFG_GLOBAL },
492         { "printmotd", sPrintMotd, SSHCFG_GLOBAL },
493         { "printlastlog", sPrintLastLog, SSHCFG_GLOBAL },
494         { "ignorerhosts", sIgnoreRhosts, SSHCFG_GLOBAL },
495         { "ignoreuserknownhosts", sIgnoreUserKnownHosts, SSHCFG_GLOBAL },
496         { "x11forwarding", sX11Forwarding, SSHCFG_ALL },
497         { "x11displayoffset", sX11DisplayOffset, SSHCFG_ALL },
498         { "x11uselocalhost", sX11UseLocalhost, SSHCFG_ALL },
499         { "xauthlocation", sXAuthLocation, SSHCFG_GLOBAL },
500         { "strictmodes", sStrictModes, SSHCFG_GLOBAL },
501         { "permitemptypasswords", sEmptyPasswd, SSHCFG_ALL },
502         { "permituserenvironment", sPermitUserEnvironment, SSHCFG_GLOBAL },
503         { "uselogin", sUseLogin, SSHCFG_GLOBAL },
504         { "compression", sCompression, SSHCFG_GLOBAL },
505         { "tcpkeepalive", sKeepAlives, SSHCFG_GLOBAL },
506         { "keepalive", sKeepAlives, SSHCFG_GLOBAL },            /* obsolete */
504         { "keepalive", sKeepAlives, SSHCFG_GLOBAL },
507         { "allowtcpforwarding", sAllowTcpForwarding, SSHCFG_ALL },
508         { "allowusers", sAllowUsers, SSHCFG_GLOBAL },
509         { "denyusers", sDenyUsers, SSHCFG_GLOBAL },
510         { "allowgroups", sAllowGroups, SSHCFG_GLOBAL },
511         { "denygroups", sDenyGroups, SSHCFG_GLOBAL },
512         { "ciphers", sCiphers, SSHCFG_GLOBAL },
513         { "macs", sMacs, SSHCFG_GLOBAL},
514         { "protocol", sProtocol,SSHCFG_GLOBAL },
515         { "gatewayports", sGatewayPorts, SSHCFG_ALL },
516         { "subsystem", sSubsystem, SSHCFG_GLOBAL},
517         { "maxstartups", sMaxStartups, SSHCFG_GLOBAL },
518         { "banner", sBanner, SSHCFG_ALL },
519         { "verifyreversemapping", sVerifyReverseMapping, SSHCFG_GLOBAL },
520         { "reversemappingcheck", sVerifyReverseMapping,SSHCFG_GLOBAL },
521         { "clientaliveinterval", sClientAliveInterval, SSHCFG_GLOBAL },
522         { "clientalivecountmax", sClientAliveCountMax, SSHCFG_GLOBAL },
523         { "authorizedkeysfile", sAuthorizedKeysFile, SSHCFG_GLOBAL },
524         { "authorizedkeysfile2", sAuthorizedKeysFile2, SSHCFG_GLOBAL },
525         { "maxauthtries", sMaxAuthTries, SSHCFG_ALL },
526         { "maxauthtrieslog", sMaxAuthTriesLog, SSHCFG_GLOBAL },
527         { "useprivilegeseparation", sUsePrivilegeSeparation, SSHCFG_GLOBAL },
528         { "lookupclienthostnames", sLookupClientHostnames, SSHCFG_GLOBAL },
529         { "useopensslengine", sUseOpenSSLEngine, SSHCFG_GLOBAL },
530         { "chrootdirectory", sChrootDirectory, SSHCFG_ALL },
531         { "preuserauthhook", sPreUserauthHook, SSHCFG_ALL},
532         { "match", sMatch, SSHCFG_ALL },
533         { "pamserviceprefix", sPAMServicePrefix, SSHCFG_GLOBAL },
534         { "pamservicename", sPAMServiceName, SSHCFG_GLOBAL },

536         { NULL, sBadOption, 0 }
537 };
______unchanged_portion_omitted_
```

```
**********************************************************
   27823 Thu Jan 17 15:17:54 2013
new/usr/src/man/man4/ssh_config.4
3477 SunSSH config should accept TCPKeepAlive as synonym for KeepAlive
Reviewed by: Jerry Jelinek <jerry@joyent.com>
**********************************************************
   1 '\" te
   2 .\" Copyright (c) 2009, Sun Microsystems, Inc. All Rights Reserved.
   3 .\" Copyright (c) 2013, Joyent, Inc. All Rights Reserved.
   4 .\" To view Portions Copyright for OpenSSH, the default path is /var/sadm/pkg/SU
   5 .\" The contents of this file are subject to the terms of the Common Development
   6 .\"  See the License for the specific language governing permissions and limitat
   7 .\" the fields enclosed by brackets "[]" replaced with your own identifying info
   8 .TH SSH_CONFIG 4 "Jan 17, 2013"
   7 .TH SSH_CONFIG 4 "Apr 20, 2009"
   9 .SH NAME
  10 ssh_config \- ssh configuration file
  11 .SH SYNOPSIS
  12 .LP
  13 .nf
  14 \fB/etc/ssh/ssh_config\fR
  15 .fi

  17 .LP
  18 .nf
  19 \fB$HOME/.ssh/config\fR
  20 .fi

  22 .SH DESCRIPTION
  23 .sp
  24 .LP
  25 The first \fBssh_config\fR path, above, provides the system-wide defaults for
  26 \fBssh\fR(1). The second version is user-specific defaults for \fBssh\fR.
  27 .sp
  28 .LP
  29 \fBssh\fR obtains configuration data from the following sources, in this order:
  30 command line options, user's configuration file (\fB$HOME/.ssh/config\fR), and
  31 system-wide configuration file (\fB/etc/ssh/ssh_config\fR). For each parameter,
  32 the first obtained value is used. The configuration files contain sections
  33 bracketed by \fBHost\fR specifications, and that section is applied only for
  34 hosts that match one of the patterns given in the specification. The matched
  35 host name is the one given on the command line.
  36 .sp
  37 .LP
  38 Since the first obtained value for each parameter is used, host-specific
  39 declarations should be given near the beginning of the file, and general
  40 defaults at the end.
  41 .sp
  42 .LP
  43 The configuration file has the following format and syntax:
  44 .RS +4
  45 .TP
  46 .ie t \(bu
  47 .el o
  48 Empty lines and lines starting with \fB#\fR are comments.
  49 .RE
  50 .RS +4
  51 .TP
  52 .ie t \(bu
  53 .el o
  54 Non-commented lines are of the form:
  55 .sp
  56 .in +2
  57 .nf
  58 \fIkeyword\fR \fIarguments\fR
  59 .fi
```



```
  60 .in -2
  61 .sp

  63 .RE
  64 .RS +4
  65 .TP
  66 .ie t \(bu
  67 .el o
  68 Configuration options can be separated by white space or optional whitespace
  69 and exactly one equal sign. The latter format allows you to avoid the need to
  70 quote white space when specifying configuration options using the \fB-o\fR
  71 option to \fBssh\fR, \fBscp\fR, and \fBsftp\fR.
  72 .RE
  73 .sp
  74 .LP
  75 The possible keywords and their meanings are listed in the following
  76 list.Keywords are case-insensitive and arguments are case-sensitive.
  77 .sp
  78 .ne 2
  79 .na
  80 \fB\fBBatchMode\fR\fR
  81 .ad
  82 .sp .6
  83 .RS 4n
  84 The argument must be \fByes\fR or \fBno\fR. If set to \fByes\fR,
  85 passphrase/password querying is disabled. This option is useful in scripts and
  86 other batch jobs where you have no user to supply the password.
  87 .RE

  89 .sp
  90 .ne 2
  91 .na
  92 \fB\fBBindAddress\fR\fR
  93 .ad
  94 .sp .6
  95 .RS 4n
  96 Specify the interface to transmit from on machines with multiple interfaces or
  97 aliased addresses. This option does not work if \fBUsePrivilegedPort\fR is set
  98 to \fByes\fR.
  99 .RE

 101 .sp
 102 .ne 2
 103 .na
 104 \fB\fBCheckHostIP\fR\fR
 105 .ad
 106 .sp .6
 107 .RS 4n
 108 If this flag is set to \fByes\fR, \fBssh\fR additionally checks the host IP
 109 address in the \fBknown_hosts\fR file. This allows \fBssh\fR to detect if a
 110 host key changed due to DNS spoofing. If the option is set to \fBno\fR, the
 111 check is not executed.
 112 .RE

 114 .sp
 115 .ne 2
 116 .na
 117 \fB\fBCipher\fR\fR
 118 .ad
 119 .sp .6
 120 .RS 4n
 121 Specifies the cipher to use for encrypting the session in protocol version 1.
 122 Only a single cipher can be specified. Currently, \fBblowfish, 3des,\fR and
 123 \fBdes\fR are supported. \fB3des\fR (triple-\fBdes\fR) is an
 124 encrypt-decrypt-encrypt triple with three different keys. It is believed to be
 125 secure. \fBblowfish\fR is a fast block cipher. It appears very secure and is
```

```
126 much faster than \fB3des\fR. \fBdes\fR is only supported in the \fBssh\fR
127 client for interoperability with legacy protocol 1 implementations that do not
128 support the \fB3des\fR cipher. Its use is strongly discouraged due to
129 cryptographic weaknesses. The default is \fB3des\fR.
130 .RE

132 .sp
133 .ne 2
134 .na
135 \fB\fBCiphers\fR\fR
136 .ad
137 .sp .6
138 .RS 4n
139 Specifies the ciphers allowed for protocol version 2 in order of preference.
140 Multiple ciphers must be comma separated.
141 .sp
142 The default cipher list contains all supported ciphers in this order:
143 .sp
144 .in +2
145 .nf
146 aes128-ctr, aes192-ctr, aes256-ctr, arcfour128, arcfour256, arcfour, aes128-cbc,
147 aes192-cbc, aes256-cbc, arcfour, 3des-cbc,blowfish-cbc
148 .fi
149 .in -2
150 .sp

152 While CBC modes are not considered as secure as other modes in connection with
153 the SSH protocol 2 they are present at the back of the default client cipher
154 list for backward compatibility with SSH servers that do not support other
155 cipher modes.
156 .RE

158 .sp
159 .ne 2
160 .na
161 \fB\fBClearAllForwardings\fR\fR
162 .ad
163 .sp .6
164 .RS 4n
165 Specifies that all local, remote, and dynamic port forwardings specified in the
166 configuration files or on the command line be cleared. This option is primarily
167 useful when used from the \fBssh\fR command line to clear port forwardings set
168 in configuration files and is automatically set by \fBscp\fR(1) and
169 \fBsftp\fR(1). The argument must be \fByes\fR or \fBno\fR. The default is
170 \fBno\fR.
171 .RE

173 .sp
174 .ne 2
175 .na
176 \fB\fBCompression\fR\fR
177 .ad
178 .sp .6
179 .RS 4n
180 Specifies whether to use compression. The argument must be \fByes\fR or
181 \fBno\fR. Defaults to \fBno\fR.
182 .RE

184 .sp
185 .ne 2
186 .na
187 \fB\fBCompressionLevel\fR\fR
188 .ad
189 .sp .6
190 .RS 4n
191 Specifies the compression level to use if compression is enabled. The argument
```



```
192 must be an integer from 1 (fast) to 9 (slow, best). The default level is 6,
193 which is good for most applications. This option applies to protocol version 1
194 only.
195 .RE

197 .sp
198 .ne 2
199 .na
200 \fB\fBConnectionAttempts\fR\fR
201 .ad
202 .sp .6
203 .RS 4n
204 Specifies the number of tries (one per second) to make before falling back to
205 \fBrsh\fR or exiting. The argument must be an integer. This can be useful in
206 scripts if the connection sometimes fails. The default is 1.
207 .RE

209 .sp
210 .ne 2
211 .na
212 \fB\fBConnectTimeout\fR\fR
213 .ad
214 .sp .6
215 .RS 4n
216 Specifies the timeout (in seconds) used when connecting to the \fBssh\fR
217 server, instead of using the default system TCP timeout. This value is used
218 only when the target is down or truly unreachable, not when it refuses the
219 connection.
220 .RE

222 .sp
223 .ne 2
224 .na
225 \fB\fBDisableBanner\fR\fR
226 .ad
227 .sp .6
228 .RS 4n
229 If set to \fByes\fR, disables the display of the banner message. If set to
230 \fBin-exec-mode\fR, disables the display of banner message when in remote
231 command mode only.
232 .sp
233 The default value is \fBno\fR, which means that the banner is displayed unless
234 the log level is \fBQUIET\fR, \fBFATAL\fR, or \fBERROR\fR. See also the
235 \fBBanner\fR option in \fBsshd_config\fR(4). This option applies to protocol
236 version 2 only.
237 .RE

239 .sp
240 .ne 2
241 .na
242 \fB\fBDynamicForward\fR\fR
243 .ad
244 .sp .6
245 .RS 4n
246 Specifies that a TCP/IP port on the local machine be forwarded over the secure
247 channel. The application protocol is then used to determine where to connect to
248 from the remote machine.
249 .sp
250 The argument must be \fB[\fR\fIbind_address\fR\fB:]\fR\fIport\fR. IPv6
251 addresses can be specified by enclosing addresses in square brackets or by
252 using an alternative syntax: \fB[\fR\fIbind_address\fR\fB/]\fR\fIport\fR. By
253 default, the local port is bound in accordance with the \fBGatewayPorts\fR
254 setting. However, an explicit \fIbind_address\fR can be used to bind the
255 connection to a specific address. The \fIbind_address\fR of \fBlocalhost\fR
256 indicates that the listening port be bound for local use only, while an empty
257 address or \fB*\fR indicates that the port should be available from all
```

```
interfaces.
.sp
Currently the \fBSOCKS4\fR and \fBSOCKS5\fR protocols are supported, and
\fBssh\fR acts as a \fBSOCKS\fR server. Multiple forwardings can be specified
and additional forwardings can be specified on the command line. Only a user
with enough privileges can forward privileged ports.
.RE

.sp
.ne 2
.na
\fB\fBEscapeChar\fR\fR
.ad
.sp .6
.RS 4n
Sets the escape character. The default is tilde (\fB~\fR). The escape character
can also be set on the command line. The argument should be a single character,
\fB^\fR, followed by a letter, or \fBnone\fR to disable the escape character
entirely (making the connection transparent for binary data).
.RE

.sp
.ne 2
.na
\fB\fBFallBackToRsh\fR\fR
.ad
.sp .6
.RS 4n
Specifies that if connecting with \fBssh\fR fails due to a connection refused
error (there is no \fBsshd\fR(1M) listening on the remote host), \fBrsh\fR(1)
should automatically be used instead (after a suitable warning about the
session being unencrypted). The argument must be \fByes\fR or \fBno\fR.
.RE

.sp
.ne 2
.na
\fB\fBForwardAgent\fR\fR
.ad
.sp .6
.RS 4n
Specifies whether the connection to the authentication agent (if any) is
forwarded to the remote machine. The argument must be \fByes\fR or \fBno\fR.
The default is \fBno\fR.
.sp
Agent forwarding should be enabled with caution. Users with the ability to
bypass file permissions on the remote host (for the agent's Unix-domain socket)
can access the local agent through the forwarded connection. An attacker cannot
obtain key material from the agent, however he can perform operations on the
keys that enable him to authenticate using the identities loaded into the
agent.
.RE

.sp
.ne 2
.na
\fB\fBForwardX11\fR\fR
.ad
.sp .6
.RS 4n
Specifies whether X11 connections are automatically redirected over the secure
channel and \fBDISPLAY\fR set. The argument must be \fByes\fR or \fBno\fR. The
default is \fBno\fR.
.sp
X11 forwarding should be enabled with caution. Users with the ability to bypass
file permissions on the remote host (for the user's X authorization database)
```



```
can access the local \fBX11\fR display through the forwarded connection. An
attacker might then be able to perform activities such as keystroke monitoring.
See the \fBForwardX11Trusted\fR option for more information how to prevent
this.
.RE

.sp
.ne 2
.na
\fB\fBForwardX11Trusted\fR\fR
.ad
.sp .6
.RS 4n
If this option is set to \fByes\fR, remote X11 clients have full access to the
original X11 display. This option is set to \fByes\fR by default.
.sp
If this option is set to \fBno\fR, remote X11 clients are considered untrusted
and prevented from stealing or tampering with data belonging to trusted X11
clients. Furthermore, the \fBxauth\fR(1) token used for the session is set to
expire after 20 minutes. Remote clients are refused access after this time.
.sp
See the X11 SECURITY extension specification for full details on the
restrictions imposed on untrusted clients.
.RE

.sp
.ne 2
.na
\fB\fBGatewayPorts\fR\fR
.ad
.sp .6
.RS 4n
Specifies whether remote hosts are allowed to connect to local forwarded ports.
By default, \fBssh\fR binds local port forwardings to the loopback address.
This prevents other remote hosts from connecting to forwarded ports.
\fBGatewayPorts\fR can be used to specify that \fBssh\fR should bind local port
forwardings to the wildcard address, thus allowing remote hosts to connect to
forwarded ports. The argument must be \fByes\fR or \fBno\fR. The default is
\fBno\fR.
.RE

.sp
.ne 2
.na
\fB\fBGlobalKnownHostsFile\fR\fR
.ad
.sp .6
.RS 4n
Specifies a file to use instead of \fB/etc/ssh/ssh_known_hosts\fR.
.RE

.sp
.ne 2
.na
\fB\fBGSSAPIAuthentication\fR\fR
.ad
.sp .6
.RS 4n
Enables/disables GSS-API user authentication. The default is \fByes\fR.
.RE

.sp
.ne 2
.na
\fB\fBGSSAPIDelegateCredentials\fR\fR
.ad
```

```
390 .sp .6
391 .RS 4n
392 Enables/disables GSS-API credential forwarding. The default is \fBno\fR.
393 .RE

395 .sp
396 .ne 2
397 .na
398 \fB\fBGSSAPIKeyExchange\fR\fR
399 .ad
400 .sp .6
401 .RS 4n
402 Enables/disables GSS-API-authenticated key exchanges. The default is \fByes\fR.
403 .sp
404 This option is intended primarily to allow users to disable the use of GSS-API
405 key exchange for SSHv2 when it would otherwise be selected and then fail (due
406 to server misconfiguration, for example). SSHv2 key exchange failure always
407 results in disconnection.
408 .sp
409 This option also enables the use of the GSS-API to authenticate the user to the
410 server after the key exchange. GSS-API key exchange can succeed but the
411 subsequent authentication using the GSS-API fail if the server does not
412 authorize the user’s GSS principal name to the target user account.
413 .RE

415 .sp
416 .ne 2
417 .na
418 \fB\fBHashKnownHosts\fR\fR
419 .ad
420 .sp .6
421 .RS 4n
422 Indicates that \fBssh\fR(1), should hash host names and addresses when they are
423 added to \fB~/.ssh/known_hosts\fR. These hashed names can be used normally by
424 \fBssh\fR(1) and \fBsshd\fR(1M), but they do not reveal identifying information
425 should the file’s contents be disclosed. The default is \fBno\fR. Existing
426 names and addresses in known hosts files are not be converted automatically,
427 but can be manually hashed using \fBssh-keygen\fR(1).
428 .RE

430 .sp
431 .ne 2
432 .na
433 \fB\fBHost\fR\fR
434 .ad
435 .sp .6
436 .RS 4n
437 Restricts the following declarations (up to the next \fBHost\fR keyword) to be
438 only for those hosts that match one of the patterns given after the keyword. An
439 asterisk (\fB*\fR) and a question mark (\fB?\fR) can be used as wildcards in
440 the patterns. A single asterisk as a pattern can be used to provide global
441 defaults for all hosts. The host is the host name argument given on the command
442 line (that is, the name is not converted to a canonicalized host name before
443 matching).
444 .RE

446 .sp
447 .ne 2
448 .na
449 \fB\fBHostbasedAuthentication\fR\fR
450 .ad
451 .sp .6
452 .RS 4n
453 Specifies whether to try \fBrhosts\fR-based authentication with public key
454 authentication. The argument must be \fByes\fR or \fBno\fR. The default is
455 \fBno\fR. This option applies to protocol version 2 only and is similar to
```



```
456 \fBRhostsRSAAuthentication\fR.
457 .RE

459 .sp
460 .ne 2
461 .na
462 \fB\fBHostKeyAlgorithms\fR\fR
463 .ad
464 .sp .6
465 .RS 4n
466 Specifies the protocol version 2 host key algorithms that the client wants to
467 use in order of preference. The default for this option is:
468 \fBssh-rsa,ssh-dss\fR.
469 .RE

471 .sp
472 .ne 2
473 .na
474 \fB\fBHostKeyAlias\fR\fR
475 .ad
476 .sp .6
477 .RS 4n
478 Specifies an alias that should be used instead of the real host name when
479 looking up or saving the host key in the host key database files. This option
480 is useful for tunneling \fBssh\fR connections or for multiple servers running
481 on a single host.
482 .RE

484 .sp
485 .ne 2
486 .na
487 \fB\fBHostName\fR\fR
488 .ad
489 .sp .6
490 .RS 4n
491 Specifies the real host name to log into. This can be used to specify nicknames
492 or abbreviations for hosts. Default is the name given on the command line.
493 Numeric IP addresses are also permitted (both on the command line and in
494 \fBHostName\fR specifications).
495 .RE

497 .sp
498 .ne 2
499 .na
500 \fB\fBIdentityFile\fR\fR
501 .ad
502 .sp .6
503 .RS 4n
504 Specifies a file from which the user’s RSA or DSA authentication identity is
505 read. The default is \fB$HOME/.ssh/identity\fR for protocol version 1 and
506 \fB$HOME/.ssh/id_rsa\fR and \fB$HOME/.ssh/id_dsa\fR for protocol version 2.
507 Additionally, any identities represented by the authentication agent is used
508 for authentication. The file name can use the tilde syntax to refer to a user’s
509 home directory. It is possible to have multiple identity files specified in
510 configuration files; all these identities is tried in sequence.
511 .RE

513 .sp
514 .ne 2
515 .na
516 \fB\fBIgnoreIfUnknown\fR\fR
517 .ad
518 .sp .6
519 .RS 4n
520 Specifies a comma-separated list of \fBssh_config\fR parameters, which, if
521 unknown to \fBssh\fR(1), are to be ignored by \fBssh\fR.
```

```
522 .sp
523 This parameter is primarily intended to be used in the per-user
524 \fBssh_config\fR, \fB~/.ssh/config\fR. While this parameter can also be used in
525 the system wide \fB/etc/ssh/ssh_config\fR file, it is generally useless as the
526 capabilities of the \fBssh\fR(1) client on that host should match that file.
527 .RE

529 .sp
530 .ne 2
531 .na
532 \fB\fBTCPKeepAlive\fR\fR
531 \fB\fBKeepAlive\fR\fR
533 .ad
534 .sp .6
535 .RS 4n
536 Specifies whether the system should send TCP keepalive messages to the other
537 side. If they are sent, death of the connection or crash of one of the machines
538 is properly noticed. However, this means that connections die if the route is
539 down temporarily, which can be a source of annoyance.
540 .sp
541 The default is \fByes\fR (to send keepalives), which means the client notices
542 if the network goes down or the remote host dies. This is important in scripts,
543 and many users want it too. To disable keepalives, the value should be set to
544 \fBno\fR in both the server and the client configuration files.
545 .RE

547 .sp
548 .ne 2
549 .na
550 \fB\fBLocalForward\fR\fR
551 .ad
552 .sp .6
553 .RS 4n
554 Specifies that a TCP/IP port on the local machine be forwarded over the secure
555 channel to a given \fIhost\fR:\fIport\fR from the remote machine. The first
556 argument must be \fB[\fR\fIbind_address\fR\fB:]\fR\fIport\fR and the second
557 must be \fIhost\fR\fB:\fR\fIport\fR. IPv6 addresses can be specified by
558 enclosing addresses in square brackets or by using an alternative syntax:
559 \fB[\fR\fIbind_address\fR\fB/]\fR\fIport\fR and \fIhost\fR\fB/\fR\fIport\fR.
560 Multiple forwardings can be specified and additional forwardings can be given
561 on the command line. Only a user with enough privileges can forward privileged
562 ports. By default, the local port is bound in accordance with the
563 \fBGatewayPorts\fR setting. However, an explicit \fIbind_address\fR can be used
564 to bind the connection to a specific address. The \fIbind_address\fR of
565 \fIlocalhost\fR indicates that the listening port be bound for local use only,
566 while an empty address or \fB*\fR indicates that the port should be available
567 from all interfaces.
568 .RE

570 .sp
571 .ne 2
572 .na
573 \fB\fBLogLevel\fR\fR
574 .ad
575 .sp .6
576 .RS 4n
577 Gives the verbosity level that is used when logging messages from \fBssh\fR.
578 The possible values are: \fBFATAL\fR, \fBERROR\fR, \fBQUIET\fR, \fBINFO\fR,
579 \fBVERBOSE\fR, \fBDEBUG\fR, \fBDEBUG1\fR, \fBDEBUG2\fR, and \fBDEBUG3\fR. The
580 default is \fBINFO\fR. \fBDEBUG\fR and \fBDEBUG1\fR are equivalent.
581 \fBDEBUG2\fR and \fBDEBUG3\fR each specify higher levels of verbose output.
582 .RE

584 .sp
585 .ne 2
586 .na
```



```
587 \fB\fBMACs\fR\fR
588 .ad
589 .sp .6
590 .RS 4n
591 Specifies the MAC (message authentication code) algorithms in order of
592 preference. The MAC algorithm is used in protocol version 2 for data integrity
593 protection. Multiple algorithms must be comma-separated. The default is
594 \fBhmac-md5,hmac-sha1,hmac-sha1-96,hmac-md5-96\fR.
595 .RE

597 .sp
598 .ne 2
599 .na
600 \fB\fBNoHostAuthenticationForLocalhost\fR\fR
601 .ad
602 .sp .6
603 .RS 4n
604 This option can be used if the home directory is shared across machines. In
605 this case \fBlocalhost\fR refers to a different machine on each of the machines
606 and the user gets many warnings about changed host keys. However, this option
607 disables host authentication for \fBlocalhost\fR. The argument to this keyword
608 must be \fByes\fR or \fBno\fR. The default is to check the host key for
609 \fBlocalhost\fR.
610 .RE

612 .sp
613 .ne 2
614 .na
615 \fB\fBNumberOfPasswordPrompts\fR\fR
616 .ad
617 .sp .6
618 .RS 4n
619 Specifies the number of attempts before giving up for password and
620 keyboard-interactive methods. Attempts for each method are counted separately.
621 The argument to this keyword must be an integer. The default is 3.
622 .RE

624 .sp
625 .ne 2
626 .na
627 \fB\fBPasswordAuthentication\fR\fR
628 .ad
629 .sp .6
630 .RS 4n
631 Specifies whether to use password authentication. The argument to this keyword
632 must be \fByes\fR or \fBno\fR. This option applies to both protocol versions 1
633 and 2. The default is \fByes\fR.
634 .RE

636 .sp
637 .ne 2
638 .na
639 \fB\fBPort\fR\fR
640 .ad
641 .sp .6
642 .RS 4n
643 Specifies the port number to connect on the remote host. The default is 22.
644 .RE

646 .sp
647 .ne 2
648 .na
649 \fB\fBPreferredAuthentications\fR\fR
650 .ad
651 .sp .6
652 .RS 4n
```

```
653 Specifies the order in which the client should try protocol 2 authentication
654 methods. This allows a client to prefer one method (for example,
655 \fBkeyboard-interactive\fR) over another method (for example, \fBpassword\fR).
656 The default for this option is:
657 \fBhostbased,publickey,keyboard-interactive,password\fR.
658 .RE

660 .sp
661 .ne 2
662 .na
663 \fB\fBProtocol\fR\fR
664 .ad
665 .sp .6
666 .RS 4n
667 Specifies the protocol versions \fBssh\fR should support in order of
668 preference. The possible values are \fB1\fR and \fB2\fR. Multiple versions must
669 be comma-separated. The default is \fB2,1\fR. This means that \fBssh\fR tries
670 version 2 and falls back to version 1 if version 2 is not available.
671 .RE

673 .sp
674 .ne 2
675 .na
676 \fB\fBProxyCommand\fR\fR
677 .ad
678 .sp .6
679 .RS 4n
680 Specifies the command to use to connect to the server. The command string
681 extends to the end of the line, and is executed with \fB/bin/sh\fR. In the
682 command string, \fB%h\fR is substituted by the host name to connect and
683 \fB%p\fR by the port. The string can be any valid command, and should read from
684 its standard input and write to its standard output. It should eventually
685 connect an \fBsshd\fR(1M) server running on some machine, or execute \fBsshd\fR
686 \fB-i\fR somewhere. Host key management is done using the \fBHostName\fR of the
687 host being connected (defaulting to the name typed by the user).
688 \fBCheckHostIP\fR is not available for connects with a proxy command.
689 .RE

691 .sp
692 .ne 2
693 .na
694 \fB\fBPubkeyAuthentication\fR\fR
695 .ad
696 .sp .6
697 .RS 4n
698 Specifies whether to try public key authentication. The argument to this
699 keyword must be \fByes\fR or \fBno\fR. The default is \fByes\fR. This option
700 applies to protocol version 2 only.
701 .RE

703 .sp
704 .ne 2
705 .na
706 \fB\fBRekeyLimit\fR\fR
707 .ad
708 .sp .6
709 .RS 4n
710 Specifies the maximum amount of data that can be transmitted before the session
711 key is renegotiated. The argument is the number of bytes, with an optional
712 suffix of \fBK\fR, \fBM\fR, or \fBG\fR to indicate Kilobytes, Megabytes, or
713 Gigabytes, respectively. The default is between \fB1G\fR and \fB4G\fR,
714 depending on the cipher. This option applies to protocol version 2 only.
715 .RE

717 .sp
718 .ne 2
```



```
719 .na
720 \fB\fBRemoteForward\fR\fR
721 .ad
722 .sp .6
723 .RS 4n
724 Specifies that a TCP/IP port on the remote machine be forwarded over the secure
725 channel to a given \fB\fIhost\fR:\fIport\fR\fR from the local machine. The
726 first argument must be \fB[\fR\fIbind_address\fR\fB:]\fR\fIport\fR and the
727 second argument must be \fIhost\fR\fB:\fR\fIport\fR. IPv6 addresses can be
728 specified by enclosing addresses in square brackets or by using an alternative
729 syntax: \fB[\fR\fIbind_address\fR\fB/]\fR\fIport\fR and
730 \fIhost\fR\fB/\fR\fIport\fR. You can specify multiple forwardings and give
731 additional forwardings on the command line. Only a user with enough privileges
732 can forward privileged ports.
733 .sp
734 If the \fIbind_address\fR is not specified, the default is to only bind to
735 loopback addresses. If the \fIbind_address\fR is \fB*\fR or an empty string,
736 then the forwarding is requested to listen on all interfaces. Specifying a
737 remote \fIbind_address\fR only succeeds if the server's \fBGatewayPorts\fR
738 option is enabled. See \fBsshd_config\fR(4).
739 .RE

741 .sp
742 .ne 2
743 .na
744 \fB\fBRhostsAuthentication\fR\fR
745 .ad
746 .sp .6
747 .RS 4n
748 Specifies whether to try \fBrhosts\fR-based authentication. This declaration
749 affects only the client side and has no effect whatsoever on security.
750 Disabling \fBrhosts\fR authentication can reduce authentication time on slow
751 connections when \fBrhosts\fR authentication is not used. Most servers do not
752 permit \fBRhostsAuthentication\fR because it is not secure (see
753 \fBRhostsRSAAuthentication\fR). The argument to this keyword must be \fByes\fR
754 or \fBno\fR. This option applies only to the protocol version 1 and requires
755 that \fBssh\fR be \fBsetuid\fR root and that \fBUsePrivilegedPort\fR be set to
756 \fByes\fR.
757 .RE

759 .sp
760 .ne 2
761 .na
762 \fB\fBRhostsRSAAuthentication\fR\fR
763 .ad
764 .sp .6
765 .RS 4n
766 Specifies whether to try \fBrhosts\fR-based authentication with RSA host
767 authentication. This is the primary authentication method for most sites. The
768 argument must be \fByes\fR or \fBno\fR. This option applies only to the
769 protocol version 1 and requires that \fBssh\fR be \fBsetuid\fR root and that
770 \fBUsePrivilegedPort\fR be set to \fByes\fR.
771 .RE

773 .sp
774 .ne 2
775 .na
776 \fB\fBServerAliveCountMax\fR\fR
777 .ad
778 .sp .6
779 .RS 4n
780 Sets the number of server alive messages which can be sent without \fBssh\fR(1)
781 receiving messages back from the server. If this threshold is reached while
782 server alive messages are being sent, \fBssh\fR disconnects from the server,
783 terminating the session. The use of server alive messages differs from
784 \fBTCPKeepAlive\fR. Server alive messages are sent through the encrypted
```

```
785 channel and are not spoofable. The TCP keep alive option enabled by
786 \fBTCPKeepAlive\fR is spoofable. The server alive mechanism is valuable when
787 the client or server depend on knowing when a connection has become inactive.
788 .sp
789 The default value is 3. If, for example, \fBServerAliveInterval\fR is set to 15
790 and \fBServerAliveCountMax\fR is left at the default, \fBssh\fR disconnects in
791 45-60 seconds if the server becomes unresponsive. This option applies to
792 protocol version 2 only.
793 .RE

795 .sp
796 .ne 2
797 .na
798 \fB\fBServerAliveInterval\fR\fR
799 .ad
800 .sp .6
801 .RS 4n
802 Sets a timeout interval in seconds after which if no data has been received
803 from the server, \fBssh\fR(1) sends a message through the encrypted channel to
804 request a response from the server. The default is 0, indicating that these
805 messages are not sent to the server. This option applies to protocol version 2
806 only.
807 .RE

809 .sp
810 .ne 2
811 .na
812 \fB\fBStrictHostKeyChecking\fR\fR
813 .ad
814 .sp .6
815 .RS 4n
816 If this flag is set to \fByes\fR, \fBssh\fR never automatically adds host keys
817 to the \fB$HOME/.ssh/known_hosts\fR file, and refuses to connect hosts whose
818 host key has changed. This provides maximum protection against trojan horse
819 attacks. However, it can be a source of inconvenience if you do not have good
820 \fB/etc/ssh/ssh_known_hosts\fR files installed and frequently connect new
821 hosts. This option forces the user to manually add any new hosts. Normally this
822 option is disabled, and new hosts are automatically added to the known host
823 files. The host keys of known hosts are verified automatically in either case.
824 The argument must be \fByes\fR or \fBno\fR or \fBask\fR. The default is
825 \fBask\fR.
826 .RE

828 .sp
829 .ne 2
830 .na
831 \fB\fBUseOpenSSLEngine\fR\fR
832 .ad
833 .sp .6
834 .RS 4n
835 Specifies whether \fBssh\fR should use the OpenSSL PKCS#11 engine for
836 offloading cryptographic operations to the Cryptographic Framework.
837 Cryptographic operations are accelerated according to the available installed
838 plug-ins. When no suitable plug-ins are present this option does not have an
839 effect. The default is \fByes\fR.
840 .RE

842 .sp
843 .ne 2
844 .na
845 \fB\fBUsePrivilegedPort\fR\fR
846 .ad
847 .sp .6
848 .RS 4n
849 Specifies whether to use a privileged port for outgoing connections. The
850 argument must be \fByes\fR or \fBno\fR. The default is \fByes\fR. Setting this
```



```
851 option to \fBno\fR turns off \fBRhostsAuthentication\fR and
852 \fBRhostsRSAAuthentication\fR. If set to \fByes\fR \fBssh\fR must be
853 \fBsetuid\fR root. Defaults to \fBno\fR.
854 .RE

856 .sp
857 .ne 2
858 .na
859 \fB\fBUser\fR\fR
860 .ad
861 .sp .6
862 .RS 4n
863 Specifies the user to log in as. This can be useful if you have different user
864 names on different machines. This saves you the trouble of having to remember
865 to enter the user name on the command line.
866 .RE

868 .sp
869 .ne 2
870 .na
871 \fB\fBUserKnownHostsFile\fR\fR
872 .ad
873 .sp .6
874 .RS 4n
875 Specifies a file to use instead of \fB$HOME/.ssh/known_hosts\fR.
876 .RE

878 .sp
879 .ne 2
880 .na
881 \fB\fBUseRsh\fR\fR
882 .ad
883 .sp .6
884 .RS 4n
885 Specifies that \fBrlogin\fR or \fBrsh\fR should be used for this host. It is
886 possible that the host does not support the \fBssh\fR protocol. This causes
887 \fBssh\fR to immediately execute \fBrsh\fR(1). All other options (except
888 \fBHostName\fR) are ignored if this has been specified. The argument must be
889 \fByes\fR or \fBno\fR.
890 .RE

892 .sp
893 .ne 2
894 .na
895 \fB\fBXAuthLocation\fR\fR
896 .ad
897 .sp .6
898 .RS 4n
899 Specifies the location of the \fBxauth\fR(1) program. The default is
900 \fB/usr/openwin/bin/xauth\fR.
901 .RE

903 .SH SEE ALSO
904 .sp
905 .LP
906 \fBrsh\fR(1), \fBssh\fR(1), \fBssh-http-proxy-connect\fR(1),
907 \fBssh-keygen\fR(1), \fBssh-socks5-proxy-connect\fR(1), \fBsshd\fR(1M),
908 \fBsshd_config\fR(4), \fBkerberos\fR(5)
909 .sp
910 .LP
911 \fIRFC 4252\fR
```

```
**********************************************************
   29080 Thu Jan 17 15:17:54 2013
new/usr/src/man/man4/sshd_config.4
3477 SunSSH config should accept TCPKeepAlive as synonym for KeepAlive
Reviewed by: Jerry Jelinek <jerry@joyent.com>
**********************************************************
   1 '\" te
   2 .\" Copyright (c) 2009, Sun Microsystems, Inc. All Rights Reserved.
   3 .\" Copyright (c) 2013, Joyent, Inc. All Rights Reserved.
   4 .\" The contents of this file are subject to the terms of the Common Development
   5 .\"  See the License for the specific language governing permissions and limitat
   6 .\" fields enclosed by brackets "[]" replaced with your own identifying informat
   7 .TH SSHD_CONFIG 4 "Jan 17, 2013"
   6 .TH SSHD_CONFIG 4 "Mar 26, 2009"
   8 .SH NAME
   9 sshd_config \- sshd configuration file
  10 .SH SYNOPSIS
  11 .LP
  12 .nf
  13 \fB/etc/ssh/sshd_config\fR
  14 .fi

  16 .SH DESCRIPTION
  17 .sp
  18 .LP
  19 The \fBsshd\fR(1M) daemon reads configuration data from
  20 \fB/etc/ssh/sshd_config\fR (or the file specified with \fBsshd\fR \fB-f\fR on
  21 the command line). The file contains keyword-value pairs, one per line. A line
  22 starting with a hash mark (\fB#\fR) and empty lines are interpreted as
  23 comments.
  24 .sp
  25 .LP
  26 The \fBsshd_config\fR file supports the following keywords. Unless otherwise
  27 noted, keywords and their arguments are case-insensitive.
  28 .sp
  29 .ne 2
  30 .na
  31 \fB\fBAllowGroups\fR\fR
  32 .ad
  33 .sp .6
  34 .RS 4n
  35 This keyword can be followed by a number of group names, separated by spaces.
  36 If specified, login is allowed only for users whose primary group or
  37 supplementary group list matches one of the patterns. Asterisk (\fB*\fR) and
  38 question mark (\fB?\fR) can be used as wildcards in the patterns. Only group
  39 names are valid; a numerical group ID is not recognized. By default, login is
  40 allowed regardless of the primary group.
  41 .RE

  43 .sp
  44 .ne 2
  45 .na
  46 \fB\fBAllowTcpForwarding\fR\fR
  47 .ad
  48 .sp .6
  49 .RS 4n
  50 Specifies whether TCP forwarding is permitted. The default is \fByes\fR.
  51 Disabling TCP forwarding does not improve security unless users are also denied
  52 shell access, as they can always install their own forwarders.
  53 .RE

  55 .sp
  56 .ne 2
  57 .na
  58 \fB\fBAllowUsers\fR\fR
  59 .ad
```



```
  60 .sp .6
  61 .RS 4n
  62 This keyword can be followed by a number of user names, separated by spaces. If
  63 specified, login is allowed only for user names that match one of the patterns.
  64 Asterisk (\fB*\fR) and question mark (\fB?\fR) can be used as wildcards in the
  65 patterns. Only user names are valid; a numerical user ID is not recognized. By
  66 default login is allowed regardless of the user name.
  67 .sp
  68 If a specified pattern takes the form \fIuser\fR@\fIhost\fR then \fIuser\fR and
  69 \fIhost\fR are checked separately, restricting logins to particular users from
  70 particular hosts.
  71 .RE

  73 .sp
  74 .ne 2
  75 .na
  76 \fB\fBAuthorizedKeysFile\fR\fR
  77 .ad
  78 .sp .6
  79 .RS 4n
  80 Specifies the file that contains the public keys that can be used for user
  81 authentication. \fBAuthorizedKeysFile\fR can contain tokens of the form
  82 \fB%T\fR, which are substituted during connection set-up. The following tokens
  83 are defined: \fB%%\fR is replaced by a literal \fB%\fR, \fB%h\fR is replaced by
  84 the home directory of the user being authenticated and \fB%u\fR is replaced by
  85 the username of that user. After expansion, \fBAuthorizedKeysFile\fR is taken
  86 to be an absolute path or one relative to the user's home directory. The
  87 default is \fB\&.ssh/authorized_keys\fR.
  88 .RE

  90 .sp
  91 .ne 2
  92 .na
  93 \fB\fBBanner\fR\fR
  94 .ad
  95 .sp .6
  96 .RS 4n
  97 In some jurisdictions, sending a warning message before authentication can be
  98 relevant for getting legal protection. The contents of the specified file are
  99 sent to the remote user before authentication is allowed. This option is only
 100 available for protocol version 2. By default, no banner is displayed.
 101 .RE

 103 .sp
 104 .ne 2
 105 .na
 106 \fB\fBChrootDirectory\fR\fR
 107 .ad
 108 .sp .6
 109 .RS 4n
 110 Specifies a path to \fBchroot\fR(2) to after authentication. This path, and all
 111 its components, must be root-owned directories that are not writable by any
 112 other user or group.
 113 .sp
 114 The server always tries to change to the user's home directory locally under
 115 the chrooted environment but a failure to do so is not considered an error. In
 116 addition, the path might contain the following tokens that are expanded at
 117 runtime once the connecting user has been authenticated: \fB%%\fR is replaced
 118 by a literal \fB%\fR, \fB%h\fR is replaced by the home directory of the user
 119 being authenticated, and \fB%u\fR is replaced by the username of that user.
 120 .sp
 121 The \fBChrootDirectory\fR must contain the necessary files and directories to
 122 support the user's session. For an interactive SSH session this requires at
 123 least a user's shell, shared libraries needed by the shell, dynamic linker, and
 124 possibly basic \fB/dev\fR nodes such as \fBnull\fR, \fBzero\fR, \fBstdin\fR,
 125 \fBstdout\fR, \fBstderr\fR, \fBrandom\fR, and \fBtty\fR. Additionally, terminal
```

databases are needed for screen oriented applications. For file transfer sessions using \fBsftp\fR with the SSH protocol version 2, no additional configuration of the environment is necessary if the in-process \fBsftp\fR server is used. See \fBSubsystem\fR for details.
.sp
The default is not to \fBchroot\fR(2).
.RE

.sp
.ne 2
.na
\fB\fBCiphers\fR\fR
.ad
.sp .6
.RS 4n
Specifies the ciphers allowed for protocol version 2. Cipher ordering on the server side is not relevant. Multiple ciphers must be comma separated.
.sp
Valid ciphers are: \fBaes128-ctr, aes192-ctr, aes256-ctr, aes128-cbc, aes192-cbc, aes256-cbc, arcfour, arcfour128, arcfour256, 3des-cbc\fR, and \fBblowfish-cbc\fR.
.sp
The default cipher list is:
.sp
.in +2
.nf
aes128-ctr,aes192-ctr,aes256-ctr,arcfour128,
arcfour256,arcfour
.fi
.in -2
.sp

Using CBC modes on the server side is not recommended due to potential security issues in connection with the SSH protocol version 2.
.RE

.sp
.ne 2
.na
\fB\fBClientAliveCountMax\fR\fR
.ad
.sp .6
.RS 4n
Sets the number of client alive messages, (see \fBClientAliveInterval\fR), that can be sent without \fBsshd\fR receiving any messages back from the client. If this threshold is reached while client alive messages are being sent, \fBsshd\fR disconnects the client, terminating the session. The use of client **alive messages is very different from \fBTCPKeepAlive\fR. The client alive** *alive messages is very different from \fBKeepAlive\fR. The client alive* messages are sent through the encrypted channel and therefore are not **spoofable. The TCP keepalive option enabled by \fBTCPKeepAlive\fR is spoofable.** *spoofable. The TCP keepalive option enabled by \fBKeepAlive\fR is spoofable.* The client alive mechanism is valuable when a client or server depend on knowing when a connection has become inactive.
.sp
The default value is 3. If \fBClientAliveInterval\fR is set to 15, and \fBClientAliveCountMax\fR is left at the default, unresponsive \fBssh\fR clients are disconnected after approximately 45 seconds.
.RE

.sp
.ne 2
.na
\fB\fBClientAliveInterval\fR\fR
.ad
.sp .6

.RS 4n
Sets a timeout interval in seconds after which, if no data has been received from the client, \fBsshd\fR sends a message through the encrypted channel to request a response from the client. The default is 0, indicating that these messages are not sent to the client. This option applies only to protocol version 2.
.RE

.sp
.ne 2
.na
\fB\fBCompression\fR\fR
.ad
.sp .6
.RS 4n
Controls whether the server allows the client to negotiate the use of compression. The default is \fByes\fR.
.RE

.sp
.ne 2
.na
\fB\fBDenyGroups\fR\fR
.ad
.sp .6
.RS 4n
Can be followed by a number of group names, separated by spaces. Users whose primary group matches one of the patterns are not allowed to log in. Asterisk (\fB*\fR) and question mark (\fB?\fR) can be used as wildcards in the patterns. Only group names are valid; a numerical group ID is not recognized. By default, login is allowed regardless of the primary group.
.RE

.sp
.ne 2
.na
\fB\fBDenyUsers\fR\fR
.ad
.sp .6
.RS 4n
Can be followed by a number of user names, separated by spaces. Login is disallowed for user names that match one of the patterns. Asterisk (\fB*\fR) and question mark (\fB?\fR) can be used as wildcards in the patterns. Only user names are valid; a numerical user ID is not recognized. By default, login is allowed regardless of the user name.
.sp
If a specified pattern takes the form \fIuser\fR@\fIhost\fR then \fIuser\fR and \fIhost\fR are checked separately, disallowing logins to particular users from particular hosts.
.RE

.sp
.ne 2
.na
\fB\fBGatewayPorts\fR\fR
.ad
.sp .6
.RS 4n
Specifies whether remote hosts are allowed to connect to ports forwarded for the client. By default, \fBsshd\fR binds remote port forwardings to the loopback address. This prevents other remote hosts from connecting to forwarded ports. \fBGatewayPorts\fR can be used to specify that \fBsshd\fR should bind remote port forwardings to the wildcard address, thus allowing remote hosts to connect to forwarded ports.
.sp
The argument can be \fBno\fR to force remote port forwardings to be available

```
256 to the local host only, \fByes\fR to force remote port forwardings to bind to
257 the wildcard address, or \fBclientspecified\fR to allow the client to select
258 the address to which the forwarding is bound. The default is \fBno\fR. See also
259 \fBRemoteForward\fR in \fBssh_config\fR(4).
260 .RE

262 .sp
263 .ne 2
264 .na
265 \fB\fBGSSAPIAuthentication\fR\fR
266 .ad
267 .sp .6
268 .RS 4n
269 Enables/disables GSS-API user authentication. The default is \fByes\fR.
270 .sp
271 Currently \fBsshd\fR authorizes client user principals to user accounts as
272 follows: if the principal name matches the requested user account, then the
273 principal is authorized. Otherwise, GSS-API authentication fails.
274 .RE

276 .sp
277 .ne 2
278 .na
279 \fB\fBGSSAPIKeyExchange\fR\fR
280 .ad
281 .sp .6
282 .RS 4n
283 Enables/disables GSS-API-authenticated key exchanges. The default is \fByes\fR.
284 .sp
285 This option also enables the use of the GSS-API to authenticate the user to
286 server after the key exchange. GSS-API key exchange can succeed but the
287 subsequent authentication using the GSS-API fail if the server does not
288 authorize the user's GSS principal name to the target user account.
289 .sp
290 Currently \fBsshd\fR authorizes client user principals to user accounts as
291 follows: if the principal name matches the requested user account, then the
292 principal is authorized. Otherwise, GSS-API authentication fails.
293 .RE

295 .sp
296 .ne 2
297 .na
298 \fB\fBGSSAPIStoreDelegatedCredentials\fR\fR
299 .ad
300 .sp .6
301 .RS 4n
302 Enables/disables the use of delegated GSS-API credentials on the server-side.
303 The default is \fByes\fR.
304 .sp
305 Specifically, this option, when enabled, causes the server to store delegated
306 GSS-API credentials in the user's default GSS-API credential store (which for
307 the Kerberos V mechanism means \fB/tmp/krb5cc_\fI<uid>\fR\fR).
308 .LP
309 Note -
310 .sp
311 .RS 2
312 \fBsshd\fR does not take any steps to explicitly destroy stored delegated
313 GSS-API credentials upon logout. It is the responsibility of PAM modules to
314 destroy credentials associated with a session.
315 .RE
316 .RE

318 .sp
319 .ne 2
320 .na
321 \fB\fBHostbasedAuthentication\fR\fR
```



```
322 .ad
323 .sp .6
324 .RS 4n
325 Specifies whether to try \fBrhosts\fR-based authentication with public key
326 authentication. The argument must be \fByes\fR or \fBno\fR. The default is
327 \fBno\fR. This option applies to protocol version 2 only and is similar to
328 \fBRhostsRSAAuthentication\fR. See \fBsshd\fR(1M) for guidelines on setting up
329 host-based authentication.
330 .RE

332 .sp
333 .ne 2
334 .na
335 \fB\fBHostbasedUsesNameFromPacketOnly\fR\fR
336 .ad
337 .sp .6
338 .RS 4n
339 Controls which hostname is searched for in the files \fB~/.shosts\fR,
340 \fB/etc/shosts.equiv\fR, and \fB/etc/hosts.equiv\fR. If this parameter is set
341 to \fByes\fR, the server uses the name the client claimed for itself and signed
342 with that host's key. If set to \fBno\fR, the default, the server uses the name
343 to which the client's IP address resolves.
344 .sp
345 Setting this parameter to \fBno\fR disables host-based authentication when
346 using NAT or when the client gets to the server indirectly through a
347 port-forwarding firewall.
348 .RE

350 .sp
351 .ne 2
352 .na
353 \fB\fBHostKey\fR\fR
354 .ad
355 .sp .6
356 .RS 4n
357 Specifies the file containing the private host key used by SSH. The default is
358 \fB/etc/ssh/ssh_host_key\fR for protocol version 1, and
359 \fB/etc/ssh/ssh_host_rsa_key\fR and \fB/etc/ssh/ssh_host_dsa_key\fR for
360 protocol version 2. \fBsshd\fR refuses to use a file if it is
361 group/world-accessible. It is possible to have multiple host key files.
362 \fBrsa1\fR keys are used for version 1 and \fBdsa\fR or \fBrsa\fR are used for
363 version 2 of the SSH protocol.
364 .RE

366 .sp
367 .ne 2
368 .na
369 \fB\fBIgnoreRhosts\fR\fR
370 .ad
371 .sp .6
372 .RS 4n
373 Specifies that \fB\&.rhosts\fR and \fB\&.shosts\fR files are not used in
374 authentication. \fB/etc/hosts.equiv\fR and \fB/etc/shosts.equiv\fR are still
375 used. The default is \fByes\fR. This parameter applies to both protocol
376 versions 1 and 2.
377 .RE

379 .sp
380 .ne 2
381 .na
382 \fB\fBIgnoreUserKnownHosts\fR\fR
383 .ad
384 .sp .6
385 .RS 4n
386 Specifies whether \fBsshd\fR should ignore the user's
387 \fB$HOME/.ssh/known_hosts\fR during \fBRhostsRSAAuthentication\fR. The default
```

```
388 is \fBno\fR. This parameter applies to both protocol versions 1 and 2.
389 .RE

391 .sp
392 .ne 2
393 .na
394 \fB\fBKbdInteractiveAuthentication\fR\fR
395 .ad
396 .sp .6
397 .RS 4n
398 Specifies whether authentication by means of the "keyboard-interactive"
399 authentication method (and PAM) is allowed. Defaults to \fByes\fR. (Deprecated:
400 this parameter can only be set to \fByes\fR.)
401 .RE

403 .sp
404 .ne 2
405 .na
406 \fB\fBTCPKeepAlive\fR\fR
405 \fB\fBKeepAlive\fR\fR
407 .ad
408 .sp .6
409 .RS 4n
410 Specifies whether the system should send keepalive messages to the other side.
411 If they are sent, death of the connection or crash of one of the machines is
412 properly noticed. However, this means that connections die if the route is down
413 temporarily, which can be an annoyance. On the other hand, if keepalives are
414 not sent, sessions can hang indefinitely on the server, leaving ghost users and
415 consuming server resources.
416 .sp
417 The default is \fByes\fR (to send keepalives), and the server notices if the
418 network goes down or the client host reboots. This avoids infinitely hanging
419 sessions.
420 .sp
421 To disable keepalives, the value should be set to \fBno\fR in both the server
422 and the client configuration files.
423 .RE

425 .sp
426 .ne 2
427 .na
428 \fB\fBKeyRegenerationInterval\fR\fR
429 .ad
430 .sp .6
431 .RS 4n
432 In protocol version 1, the ephemeral server key is automatically regenerated
433 after this many seconds (if it has been used). The purpose of regeneration is
434 to prevent decrypting captured sessions by later breaking into the machine and
435 stealing the keys. The key is never stored anywhere. If the value is 0, the key
436 is never regenerated. The default is 3600 (seconds).
437 .RE

439 .sp
440 .ne 2
441 .na
442 \fB\fBListenAddress\fR\fR
443 .ad
444 .sp .6
445 .RS 4n
446 Specifies what local address \fBsshd\fR should listen on. The following forms
447 can be used:
448 .sp
449 .in +2
450 .nf
451 ListenAddress \fIhost\fR|\fIIPv4_addr\fR|\fIIPv6_addr\fR
452 ListenAddress \fIhost\fR|\fIIPv4_addr\fR:\fIport\fR
```



```
453 ListenAddress [\fIhost\fR|\fIIPv6_addr\fR]:\fIport\fR
454 .fi
455 .in -2

457 If \fIport\fR is not specified, \fBsshd\fR listens on the address and all prior
458 \fBPort\fR options specified. The default is to listen on all local addresses.
459 Multiple \fBListenAddress\fR options are permitted. Additionally, any
460 \fBPort\fR options must precede this option for non-port qualified addresses.
461 .sp
462 The default is to listen on all local addresses. Multiple options of this type
463 are permitted. Additionally, the \fBPorts\fR options must precede this option.
464 .RE

466 .sp
467 .ne 2
468 .na
469 \fB\fBLoginGraceTime\fR\fR
470 .ad
471 .sp .6
472 .RS 4n
473 The server disconnects after this time (in seconds) if the user has not
474 successfully logged in. If the value is 0, there is no time limit. The default
475 is 120 (seconds).
476 .RE

478 .sp
479 .ne 2
480 .na
481 \fB\fBLogLevel\fR\fR
482 .ad
483 .sp .6
484 .RS 4n
485 Gives the verbosity level that is used when logging messages from \fBsshd\fR.
486 The possible values are: \fBQUIET\fR, \fBFATAL\fR, \fBERROR\fR, \fBINFO\fR,
487 \fBVERBOSE\fR, \fBDEBUG\fR, \fBDEBUG1\fR, \fBDEBUG2\fR, and \fBDEBUG3\fR. The
488 default is \fBINFO\fR. DEBUG2 and DEBUG3 each specify higher levels of
489 debugging output. Logging with level \fBDEBUG\fR violates the privacy of users
490 and is not recommended.
491 .RE

493 .sp
494 .ne 2
495 .na
496 \fB\fBLookupClientHostnames\fR\fR
497 .ad
498 .sp .6
499 .RS 4n
500 Specifies whether or not to lookup the names of client’s addresses. Defaults to
501 yes.
502 .RE

504 .sp
505 .ne 2
506 .na
507 \fBMACs\fR
508 .ad
509 .sp .6
510 .RS 4n
511 Specifies the available MAC (message authentication code) algorithms. The MAC
512 algorithm is used in protocol version 2 for data integrity protection. Multiple
513 algorithms must be comma-separated. The default is
514 \fBhmac-md5,hmac-sha1,hmac-sha1-96,hmac-md5-96\fR.
515 .RE

517 .sp
518 .ne 2
```

```
519 .na
520 \fB\fBMaxStartups\fR\fR
521 .ad
522 .sp .6
523 .RS 4n
524 Specifies the maximum number of concurrent unauthenticated connections to the
525 \fBsshd\fR daemon. Additional connections are dropped until authentication
526 succeeds or the \fBLoginGraceTime\fR expires for a connection. The default is
527 \fB10\fR.
528 .sp
529 Alternatively, random early drop can be enabled by specifying the three
530 colon-separated values \fB\fIstart\fR:\fIrate\fR:\fIfull\fR\fR (for example,
531 \fB10:30:60\fR). Referring to this example, \fBsshd\fR refuse connection
532 attempts with a probability of \fIrate\fR/100 (30% in our example) if there are
533 currently 10 (from the \fIstart\fR field) unauthenticated connections. The
534 probability increases linearly and all connection attempts are refused if the
535 number of unauthenticated connections reaches \fIfull\fR (60 in our example).
536 .RE

538 .sp
539 .ne 2
540 .na
541 \fB\fBPasswordAuthentication\fR\fR
542 .ad
543 .sp .6
544 .RS 4n
545 Specifies whether password authentication is allowed. The default is \fByes\fR.
546 This option applies to both protocol versions 1 and 2.
547 .RE

549 .sp
550 .ne 2
551 .na
552 \fB\fBPermitEmptyPasswords\fR\fR
553 .ad
554 .sp .6
555 .RS 4n
556 When password or keyboard-interactive authentication is allowed, it specifies
557 whether the server allows login to accounts with empty password strings.
558 .sp
559 If not set then the \fB/etc/default/login\fR \fBPASSREQ\fR value is used
560 instead.
561 .sp
562 \fBPASSREQ=no\fR is equivalent to \fBPermitEmptyPasswords yes\fR.
563 \fBPASSREQ=yes\fR is equivalent to \fBPermitEmptyPasswords no\fR. If neither
564 \fBPermitEmptyPasswords\fR or \fBPASSREQ\fR are set the default is \fBno\fR.
565 .RE

567 .sp
568 .ne 2
569 .na
570 \fB\fBPermitRootLogin\fR\fR
571 .ad
572 .sp .6
573 .RS 4n
574 Specifies whether the root can log in using \fBssh\fR(1). The argument must be
575 \fByes\fR, \fBwithout-password\fR, \fBforced-commands-only\fR, or \fBno\fR.
576 \fBwithout-password\fR means that root cannot be authenticated using the
577 "password" or "keyboard-interactive" methods (see description of
578 \fBKbdInteractiveAuthentication\fR). \fBforced-commands-only\fR means that
579 authentication is allowed only for \fBpublickey\fR (for SSHv2, or RSA, for
580 SSHv1) and only if the matching \fBauthorized_keys entry\fR for root has a
581 \fBcommand=\fR\fI<cmd>\fR option.
582 .sp
583 In Solaris, the default \fB/etc/ssh/sshd_config\fR file is shipped with
584 \fBPermitRootLogin\fR set to \fBno\fR. If unset by the administrator, then
```



```
585 \fBCONSOLE\fR parameter from \fB/etc/default/login\fR supplies the default
586 value as follows: if the \fBCONSOLE\fR parameter is not commented out (it can
587 even be empty, that is, "\fBCONSOLE=\fR"), then \fBwithout-password\fR is used
588 as default value. If \fBCONSOLE\fR is commented out, then the default for
589 \fBPermitRootLogin\fR is \fByes\fR.
590 .sp
591 The \fBwithout-password\fR and \fBforced-commands-only\fR settings are useful
592 for, for example, performing remote administration and backups using trusted
593 public keys for authentication of the remote client, without allowing access to
594 the root account using passwords.
595 .RE

597 .sp
598 .ne 2
599 .na
600 \fB\fBPermitUserEnvironment\fR\fR
601 .ad
602 .sp .6
603 .RS 4n
604 Specifies whether a user's \fB~/.ssh/environment\fR on the server side and
605 \fBenvironment\fR options in the \fBAuthorizedKeysFile\fR file are processed by
606 \fBsshd\fR. The default is \fBno\fR. Enabling environment processing can enable
607 users to bypass access restrictions in some configurations using mechanisms
608 such as \fBLD_PRELOAD\fR.
609 .sp
610 Environment setting from a relevant entry in \fBAuthorizedKeysFile\fR file is
611 processed only if the user was authenticated using the public key
612 authentication method. Of the two files used, values of variables set in
613 \fB~/.ssh/environment\fR are of higher priority.
614 .RE

616 .sp
617 .ne 2
618 .na
619 \fB\fBPidFile\fR\fR
620 .ad
621 .sp .6
622 .RS 4n
623 Allows you to specify an alternative to \fB/var/run/sshd.pid\fR, the default
624 file for storing the PID of the \fBsshd\fR listening for connections. See
625 \fBsshd\fR(1M).
626 .RE

628 .sp
629 .ne 2
630 .na
631 \fB\fBPort\fR\fR
632 .ad
633 .sp .6
634 .RS 4n
635 Specifies the port number that \fBsshd\fR listens on. The default is 22.
636 Multiple options of this type are permitted. See also \fBListenAddress\fR.
637 .RE

639 .sp
640 .ne 2
641 .na
642 \fB\fBPrintLastLog\fR\fR
643 .ad
644 .sp .6
645 .RS 4n
646 Specifies whether \fBsshd\fR should display the date and time when the user
647 last logged in. The default is \fByes\fR.
648 .RE

650 .sp
```

```
.ne 2
.na
\fB\fBPrintMotd\fR\fR
.ad
.sp .6
.RS 4n
Specifies whether \fBsshd\fR should display the contents of \fB/etc/motd\fR
when a user logs in interactively. (On some systems it is also displayed by the
shell or a shell startup file, such as \fB/etc/profile\fR.) The default is
\fByes\fR.
.RE

.sp
.ne 2
.na
\fB\fBProtocol\fR\fR
.ad
.sp .6
.RS 4n
Specifies the protocol versions \fBsshd\fR should support in order of
preference. The possible values are \fB1\fR and \fB2\fR. Multiple versions must
be comma-separated. The default is \fB2,1\fR. This means that \fBssh\fR tries
version 2 and falls back to version 1 if version 2 is not available.
.RE

.sp
.ne 2
.na
\fB\fBPubkeyAuthentication\fR\fR
.ad
.sp .6
.RS 4n
Specifies whether public key authentication is allowed. The default is
\fByes\fR. This option applies to protocol version 2 only.
.RE

.sp
.ne 2
.na
\fB\fBRhostsAuthentication\fR\fR
.ad
.sp .6
.RS 4n
Specifies whether authentication using \fBrhosts\fR or \fB/etc/hosts.equiv\fR
files is sufficient. Normally, this method should not be permitted because it
is insecure. \fBRhostsRSAAuthentication\fR should be used instead, because it
performs RSA-based host authentication in addition to normal \fBrhosts\fR or
\fB/etc/hosts.equiv\fR authentication. The default is \fBno\fR. This parameter
applies only to protocol version 1.
.RE

.sp
.ne 2
.na
\fB\fBRhostsRSAAuthentication\fR\fR
.ad
.sp .6
.RS 4n
Specifies whether \fBrhosts\fR or \fB/etc/hosts.equiv\fR authentication
together with successful RSA host authentication is allowed. The default is
\fBno\fR. This parameter applies only to protocol version 1.
.RE

.sp
.ne 2
.na
```



```
\fB\fBRSAAuthentication\fR\fR
.ad
.sp .6
.RS 4n
Specifies whether pure RSA authentication is allowed. The default is \fByes\fR.
This option applies to protocol version 1 only.
.RE

.sp
.ne 2
.na
\fB\fBServerKeyBits\fR\fR
.ad
.sp .6
.RS 4n
Defines the number of bits in the ephemeral protocol version 1 server key. The
minimum value is 512, and the default is 768.
.RE

.sp
.ne 2
.na
\fB\fBStrictModes\fR\fR
.ad
.sp .6
.RS 4n
Specifies whether \fBsshd\fR should check file modes and ownership of the
user's files and home directory before accepting login. This is normally
desirable because novices sometimes accidentally leave their directory or files
world-writable. The default is \fByes\fR.
.RE

.sp
.ne 2
.na
\fB\fBSubsystem\fR\fR
.ad
.sp .6
.RS 4n
Configures an external subsystem (for example, a file transfer daemon).
Arguments should be a subsystem name and a command to execute upon subsystem
request. The command \fBsftp-server\fR(1M) implements the \fBsftp\fR file
transfer subsystem.
.sp
Alternately, the name \fBinternal-sftp\fR implements an in-process \fBsftp\fR
server. This can simplify configurations using \fBChrootDirectory\fR to force a
different filesystem root on clients.
.sp
By default, no subsystems are defined. This option applies to protocol version
2 only.
.RE

.sp
.ne 2
.na
\fB\fBSyslogFacility\fR\fR
.ad
.sp .6
.RS 4n
Gives the facility code that is used when logging messages from \fBsshd\fR. The
possible values are: \fBDAEMON\fR, \fBUSER\fR, \fBAUTH\fR, \fBLOCAL0\fR,
\fBLOCAL1\fR, \fBLOCAL2\fR, \fBLOCAL3\fR, \fBLOCAL4\fR, \fBLOCAL5\fR,
\fBLOCAL6\fR, and \fBLOCAL7\fR. The default is \fBAUTH\fR.
.RE

.sp
```

```
783 .ne 2
784 .na
785 \fB\fBUseOpenSSLEngine\fR\fR
786 .ad
787 .sp .6
788 .RS 4n
789 Specifies whether \fBsshd\fR should use the OpenSSL PKCS#11 engine for
790 offloading cryptographic operations to the Cryptographic Framework.
791 Cryptographic operations are accelerated according to the available installed
792 plug-ins. When no suitable plug-ins are present this option does not have an
793 effect. The default is \fByes\fR.
794 .RE

796 .sp
797 .ne 2
798 .na
799 \fB\fBVerifyReverseMapping\fR\fR
800 .ad
801 .sp .6
802 .RS 4n
803 Specifies whether \fBsshd\fR should try to verify the remote host name and
804 check that the resolved host name for the remote IP address maps back to the
805 very same IP address. (A \fByes\fR setting means "verify".) Setting this
806 parameter to \fBno\fR can be useful where DNS servers might be down and thus
807 cause \fBsshd\fR to spend much time trying to resolve the client's IP address
808 to a name. This feature is useful for Internet-facing servers. The default is
809 \fBno\fR.
810 .RE

812 .sp
813 .ne 2
814 .na
815 \fB\fBX11DisplayOffset\fR\fR
816 .ad
817 .sp .6
818 .RS 4n
819 Specifies the first display number available for \fBsshd\fR's X11 forwarding.
820 This prevents \fBsshd\fR from interfering with real X11 servers. The default is
821 10.
822 .RE

824 .sp
825 .ne 2
826 .na
827 \fB\fBX11Forwarding\fR\fR
828 .ad
829 .sp .6
830 .RS 4n
831 Specifies whether X11 forwarding is permitted. The default is \fByes\fR.
832 Disabling X11 forwarding does not improve security in any way, as users can
833 always install their own forwarders.
834 .sp
835 When X11 forwarding is enabled, there can be additional exposure to the server
836 and to client displays if the \fBsshd\fR proxy display is configured to listen
837 on the wildcard address (see \fBX11UseLocalhost\fR). However, this is not the
838 default. Additionally, the authentication spoofing and authentication data
839 verification and substitution occur on the client side. The security risk of
840 using X11 forwarding is that the client's X11 display server can be exposed to
841 attack when the \fBssh\fR client requests forwarding (see the warnings for
842 \fBForwardX11\fR in \fBssh_config\fR(4)). A system administrator who wants to
843 protect clients that expose themselves to attack by unwittingly requesting X11
844 forwarding, should specify a \fBno\fR setting.
845 .sp
846 Disabling X11 forwarding does not prevent users from forwarding X11 traffic, as
847 users can always install their own forwarders.
848 .RE
```



```
850 .sp
851 .ne 2
852 .na
853 \fB\fBX11UseLocalhost\fR\fR
854 .ad
855 .sp .6
856 .RS 4n
857 Specifies whether \fBsshd\fR should bind the X11 forwarding server to the
858 loopback address or to the wildcard address. By default, \fBsshd\fR binds the
859 forwarding server to the loopback address and sets the hostname part of the
860 \fBDISPLAY\fR environment variable to \fBlocalhost\fR. This prevents remote
861 hosts from connecting to the proxy display. However, some older X11 clients
862 might not function with this configuration. \fBX11UseLocalhost\fR can be set to
863 \fBno\fR to specify that the forwarding server should be bound to the wildcard
864 address. The argument must be \fByes\fR or \fBno\fR. The default is \fByes\fR.
865 .RE

867 .sp
868 .ne 2
869 .na
870 \fB\fBXAuthLocation\fR\fR
871 .ad
872 .sp .6
873 .RS 4n
874 Specifies the location of the \fBxauth\fR(1) program. The default is
875 \fB/usr/X11/bin/xauth\fR and \fBsshd\fR attempts to open it when X11 forwarding
876 is enabled.
877 .RE

879 .SS "Time Formats"
880 .sp
881 .LP
882 \fBsshd\fR command-line arguments and configuration file options that specify
883 time can be expressed using a sequence of the form:
884 \fItime\fR[\fIqualifier\fR,] where \fItime\fR is a positive integer value and
885 \fIqualifier\fR is one of the following:
886 .sp
887 .ne 2
888 .na
889 \fB\fI<none>\fR\fR
890 .ad
891 .RS 10n
892 seconds
893 .RE

895 .sp
896 .ne 2
897 .na
898 \fB\fBs\fR | \fBS\fR\fR
899 .ad
900 .RS 10n
901 seconds
902 .RE

904 .sp
905 .ne 2
906 .na
907 \fB\fBm\fR | \fBM\fR\fR
908 .ad
909 .RS 10n
910 minutes
911 .RE

913 .sp
914 .ne 2
```

```
 915 .na
 916 \fB\fBh\fR | \fBH\fR\fR
 917 .ad
 918 .RS 10n
 919 hours
 920 .RE

 922 .sp
 923 .ne 2
 924 .na
 925 \fB\fBd\fR | \fBD\fR\fR
 926 .ad
 927 .RS 10n
 928 days
 929 .RE

 931 .sp
 932 .ne 2
 933 .na
 934 \fB\fBw\fR | \fB\fR\fR
 935 .ad
 936 .RS 10n
 937 weeks
 938 .RE

 940 .sp
 941 .LP
 942 Each element of the sequence is added together to calculate the total time
 943 value. For example:
 944 .sp
 945 .ne 2
 946 .na
 947 \fB\fB600\fR\fR
 948 .ad
 949 .RS 9n
 950 600 seconds (10 minutes)
 951 .RE

 953 .sp
 954 .ne 2
 955 .na
 956 \fB\fB10m\fR\fR
 957 .ad
 958 .RS 9n
 959 10 minutes
 960 .RE

 962 .sp
 963 .ne 2
 964 .na
 965 \fB\fB1h30m\fR\fR
 966 .ad
 967 .RS 9n
 968 1 hour, 30 minutes (90 minutes)
 969 .RE

 971 .SH FILES
 972 .sp
 973 .ne 2
 974 .na
 975 \fB\fB/etc/ssh/sshd_config\fR\fR
 976 .ad
 977 .RS 24n
 978 Contains configuration data for \fBsshd\fR. This file should be writable by
 979 root only, but it is recommended (though not necessary) that it be
 980 world-readable.
```



```
 981 .RE

 983 .SH ATTRIBUTES
 984 .sp
 985 .LP
 986 See \fBattributes\fR(5) for descriptions of the following attributes:
 987 .sp

 989 .sp
 990 .TS
 991 box;
 992 c | c
 993 l | l .
 994 ATTRIBUTE TYPE  ATTRIBUTE VALUE
 995 _
 996 Interface Stability     Uncommitted
 997 .TE

 999 .SH SEE ALSO
1000 .sp
1001 .LP
1002 \fBlogin\fR(1), \fBsshd\fR(1M), \fBchroot\fR(2), \fBssh_config\fR(4),
1003 \fBattributes\fR(5), \fBkerberos\fR(5)
1004 .SH AUTHORS
1005 .sp
1006 .LP
1007 OpenSSH is a derivative of the original and free \fBssh\fR 1.2.12 release by
1008 Tatu Ylonen. Aaron Campbell, Bob Beck, Markus Friedl, Niels Provos, Theo de
1009 Raadt, and Dug Song removed many bugs, re-added recent features, and created
1010 OpenSSH. Markus Friedl contributed the support for SSH protocol versions 1.5
1011 and 2.0. Niels Provos and Markus Friedl contributed support for privilege
1012 separation.
```