OpenVPN community report

December 2010

Samuli Seppänen

Table of Contents

Introduction

Community-driven OSS development produces large amounts of numeric data that can be analyzed. This is because the majority of communication takes place in open communication channels such as IRC channels, mailing lists and forums. The sources used in this report are:

- *openvpn-user mailing list archive (http://dir.gmane.org/gmane.network.openvpn.user)*
- *openvpn-devel mailing list archive* (http://dir.gmane.org/gmane.network.openvpn.devel)
- *#openvpn IRC channel statistics* (http://www.secure-computing.net/logs/openvpn.html)
- *#openvpn-devel IRC channel statistics* (http://www.secure-computing.net/logs/openvpn-devel.html)
- *OpenVPN forums statistics* (https://forums.openvpn.net/memberlist.php)
- Git distributed version control statistics (https://community.openvpn.net/openvpn/wiki/TesterDocumentation)

Although ample numeric data is available, looking at plain numbers only can easily lead to wrong conclusions. For example, a patch that fixes a bug requires that

- somebody found the bug
- somebody fixed the bug
- somebody tested the fix

Each of these steps takes time. The entire process may take minutes or days, depending on the problem.

Also, depending on developer's working practices one patch may contain full workday's worth of work, or much less. Or a complete feature, or a part of it. Similarly, the number of lines written to an IRC channel does not tell us how many of those lines were written to help some user out, and how many were just generic chit chat. So, numeric data cannot be analyzed reliably in isolation. It's necessary to understand the people and processes that are involved.

OpenVPN project during last 12 months

Starting point: where were we?

In late 2009 the OpenVPN project was managed by it's founder, James Yonan. The project already had a long history (2003-2009), so it already had a strong community around it. However, the founding of OpenVPN Technologies, Inc. and the time required to create commercial, OpenVPN-based products made it very difficult for James to run the OSS project. This is clearly visible from mailing list archives.[1] For example, many of the patches sent to the *openvpn-devel* mailing list were not merged into OpenVPN. This forced people to create external development trees and to maintain their own patchsets, without being able to work efficiently together.[2] This also created unnecessary friction

1 See http://search.gmane.org/?query=&author=james+yonan&group=gmane.network.openvpn.devel&sort=date&DEFAULTOP=and&xFILTERS=Gnetwork.openvpn.devel-Ajames%40openvpn.net---A

2 See https://community.openvpn.net/openvpn/wiki/RelatedProjects

between the company and the OpenVPN community.

Similarly, many community services – most notably the wiki[3], forums[4] and the IRC channel[5] - were maintained on external servers by community members. Additionally, the large German-speaking userbase had found a home in the OpenVPN.eu site.[6]

The change to the new, community-driven development model shows clearly in *openvpn-devel* mailinglist traffic levels[7]:

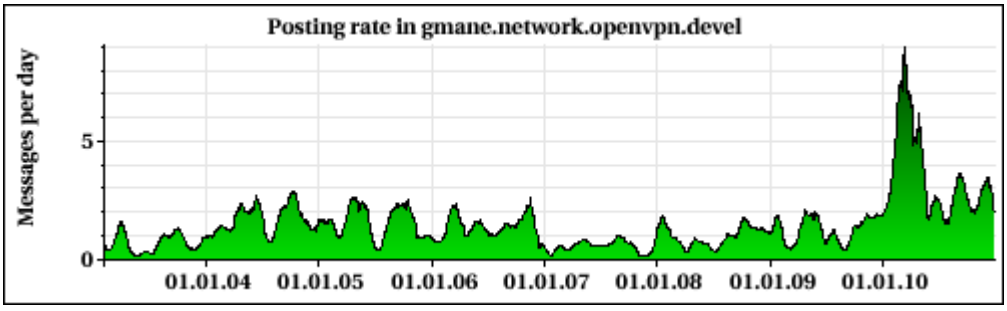

Illustration 1: openvpn-devel mailing list traffic history

The peak in January/February 2010 was caused by merging of the large number of community patches that were not yet included in OpenVPN. Traffic levels seem to have settled down now, but are still roughly twice as large as in 2008 or 2009. This traffic increase is a natural consequence of the communication requirements of the new open development model.

Development process

OpenVPN's new community-driven development process started in January 2010 with the help of OpenVPN Technologies, Inc.'s newly hired community manager, Samuli Seppänen. David Sommerseth, author of the *Eurephia authentication plugin*[8] was chosen as the leader for the ”OpenVPN testing” tree. Originally the intention was that James Yonan, original author of OpenVPN would make the stable releases. However, later David's ”testing” tree became the official development tree; this made development workflow much simpler. Currently ”testing” tree aggregates changes from several sources:

3 http://www.secure-computing.net/wiki/index.php/OpenVPN
4 http://ovpnforums.com
5 #openvpn on irc.freenode.net
6 http://www.openvpn.eu/
7 http://dir.gmane.org/gmane.network.openvpn.devel
8 http://www.eurephia.net/

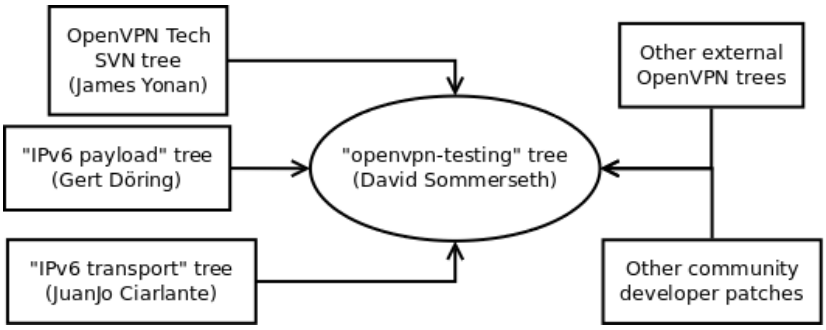

Illustration 2: Sources for new OpenVPN code

As can be seen from illustration 2, the internal OpenVPN Technologies, Inc. SVN tree used by James Yonan is just one of the development trees used in OpenVPN development.

The distributed version control system used by OpenVPN (Git) is way more powerful with branches than non-distributed VCSes (e.g. SVN). Therefore the OpenVPN ”testing” tree contains several branches used for different purposes. Additionally, branches pull changes from other branches. The most important branches are the following:

- *allmerged:* contains changes from almost all other branches
- *beta2.2:* used as a release branch for OpenVPN 2.2, which is currently at 2.2-beta5. Contains changes from *bugfix2.1*, *feat_ipv6_wintap* and *feat_misc* branches.
- *bugfix2.1:* contains only bugfixes which are considered stable.
- *feat_eurephia:* feature branch with the *Eurephia authentication plugin*
- *feat_ipv6_payload:* feature branch with the ipv6 payload support[9]
- *feat_ipv6_transport:*feature branch with the ipv6 transport support[10]
- *feat_ipv6_wintap:* feature branch with ipv6 support for Windows TAP driver
- *feat_misc:* feature branch containing several small features
- *feat_passtos:* feature branch with support for using payload's TOS (Type of Service) value for OpenVPN-encapsulated packet's TOS value[11]
- *feat_vlan_tagging:* feature branch with support for VLAN tagging[12]

Use of separate feature branches makes testing single features easier, as well as allows selective inclusion of features into each release.

Patches that go to any branch in OpenVPN ”testing” tree have to go through a *review process* before being accepted.[13] This helps ensure that all new code is of good quality and follows the existing coding standards. The review process also prevents rarely used – or entirely unnecessary – features from being integrated into OpenVPN. Additionally, developers are expected to take responsibility for their own

9 Original location: http://www.greenie.net/ipv6/openvpn.html
10 Original location: http://github.com/jjo/openvpn-ipv6/
11 Original location: http://www.caspur.it/~guerri/hacks.html
12 See http://en.wikipedia.org/wiki/VLAN_Tagging
13 Gert Döring's and James Yonan's changes are an exception to this rule, as they can be trusted to not to disappear.

features: ”drive-by” feature patches are not accepted at all.

Due to this review process there are some patches and patchsets that are *queued* and not yet integrated into OpenVPN, usually due to lack of an ACK (=approval) from developers, but sometimes due to lack of testing. The OpenVPN development processes are described in more detail here:

- https://community.openvpn.net/openvpn/wiki/DeveloperDocumentation

Summary of code changes

During 2010 several new *major features* have been integrated or are in process of being integrated into OpenVPN:

- IPv6 transport and payload support (JuanJo Ciarlante and Gert Döring)
- Automated IPv4/IPv6 connection tests for regression testing (Gert Döring)
- VLAN tagging support[14]; *awaiting testing* (Fabian Knittel)
- Eurephia authentication plugin integration (David Sommerseth)
- SSL layer abstraction / PolarSSL support for better FIPS compliance; *awaiting approval and testing*[15] (Adriaan de Jong)
- New Python-based buildsystem (James Yonan)
- New OpenVPN-GUI for Windows[16] (Heiko Hund)

There were also a large number of *small features*, most of which are included in *feat_misc* git branch:

- Allow different field in X509 to be username (Emilien Mantel)
- Use extv3 extensions such as subjectAltName as common_name; *awaiting approval* (Markus Kötter)
- Floating TLS support[17]; *awaiting approval* (Blaise Gassend)
- Allow 'lport 0' setup for random port binding (Enrico Scholz)
- MacOS X keychain support; awaiting testing[18] (Brain Raderman)
- Added --proto-force directive (James Yonan)
- Added --register-dns option for Windows (James Yonan)
- Implement challenge/response authentication support in client mode, where credentials are entered from stdin (James Yonan)
- Implemented a key/value auth channel from client to server (James Yonan)
- Implemented http-proxy-override and http-proxy-fallback directives (James Yonan)

14 http://thread.gmane.org/gmane.network.openvpn.devel/3489
15 http://thread.gmane.org/gmane.network.openvpn.devel/4199
16 http://sourceforge.net/projects/openvpn-gui/
17 http://thread.gmane.org/gmane.network.openvpn.devel/4213
18 http://thread.gmane.org/gmane.network.openvpn.devel/3631

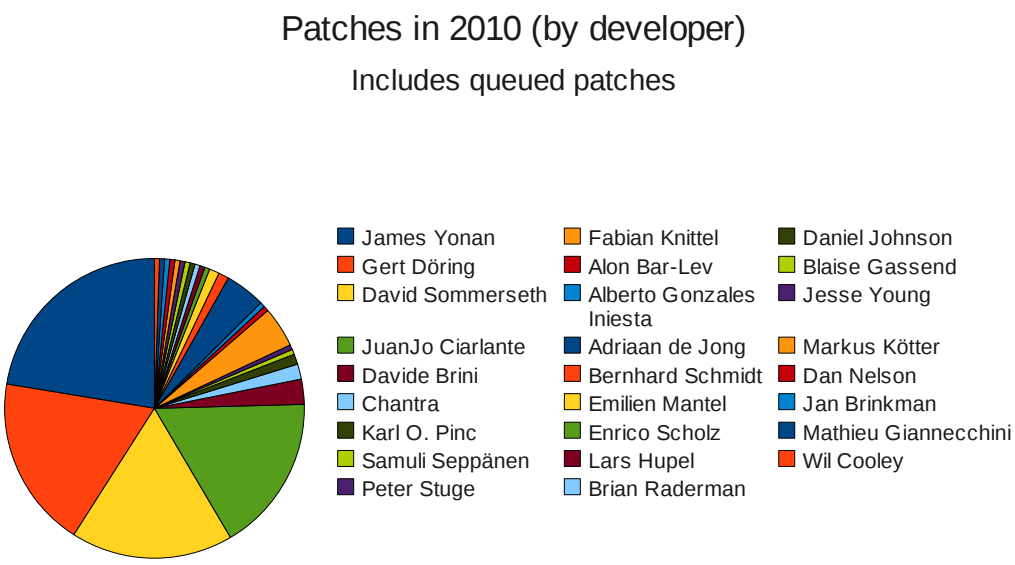

Illustration 3: Patches in 2010 (by developer)

There were a total of 183 patches merged into Git and ~20 were queued for inclusion. Some patches, such as the Eurephia patch, add a major feature. The majority of the changes not listed above were relatively small fixes and enhancements.

Developer activity

When looking at *developer activity* in the OpenVPN project, three different metrics are easy to come by:

- number of patches (see above)
- numbers of mails sent to openvpn-devel mailing list
- number of lines written to #openvpn-devel IRC channel

These metrics measure *concrete results* (patches) and *participation in the development process* (communication). See illustrations 3, 4 and 5 for details.

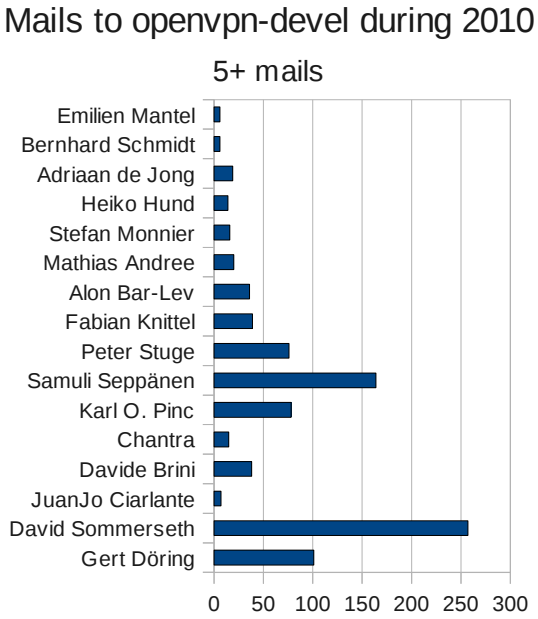

Illustration 4: Mails to openvpn-devel during 2010

User activity

There are a few useful metrics for analyzing *user activity:*

- number of mails to *openvpn-user* mailing list
- number of posts to OpenVPN forums
- number of written lines to *#openvpn* IRC channel
- number of wiki contributions
- number of bug reports

First three of the above measure the amount of *end-user support* a person gives. The third one (IRC) also has a significant social function which distorts the figures. Due to small amount of wiki contributions and bug reports per person only the *overall figures* are of any use and will even those won't be analyzed here.

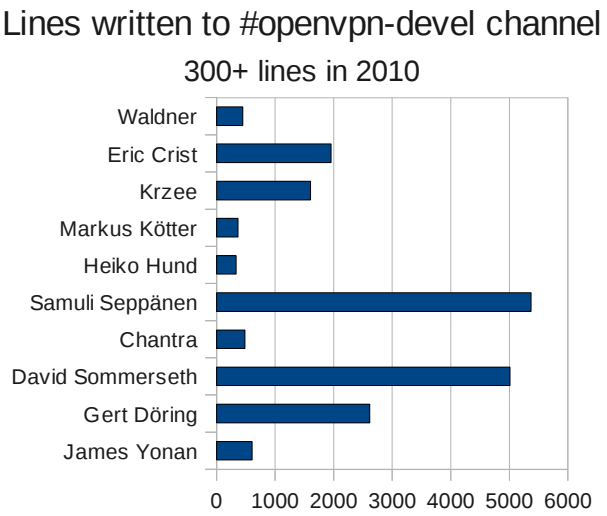

Illustration 5: Lines written to #openvpn-devel channel

Full IRC statistics are available at http://www.secure-computing.net/logs/openvpn.html.

Infrastructure & integration work

During last year the community infrastructure supporting both users and developers has been extended, cleaned up and enhanced. Many of the servers are maintained jointly by OpenVPN Technologies, Inc. employees and active community members such as Eric Crist and Krzee. The current community infrastructure consists of several services:

- LDAP self-service registration webapp (https://community.openvpn.net/account)
- Trac bug tracker and wiki (https://community.openvpn.net)
- PhpBB forums (https://forums.openvpn.net)
- Continuous integration server (buildbot) and several s.c. buildslaves associated with it
- Public test server used to verify that basic OpenVPN connectivity works
- Windows XP build computer used with the new Python-based buildsystem

All these services can be used with the same credentials. In addition, we have several external services, most important ones being:

- Git version control repository is hosted externally at SF.net

  Mailing lists (openvpn-devel, -users, -announce, -commits, and -builds) hosted on SF.net

With the exception of *mailing lists*, all services used in 2009 and earlier have been replaced with these new services. These obsoleted services were:

- SF.net bug tracker
- SF.net forums
- SF.net CVS repository
- ovpnforum.com forums

Full patch list

This list contains patches merged into “allmerged” branch in Git during 2010. All duplicates caused by cross-branch merges have been removed. None of the queued patches (~20) by Fabian Knittel, Adriaan de Jong and others have been included.

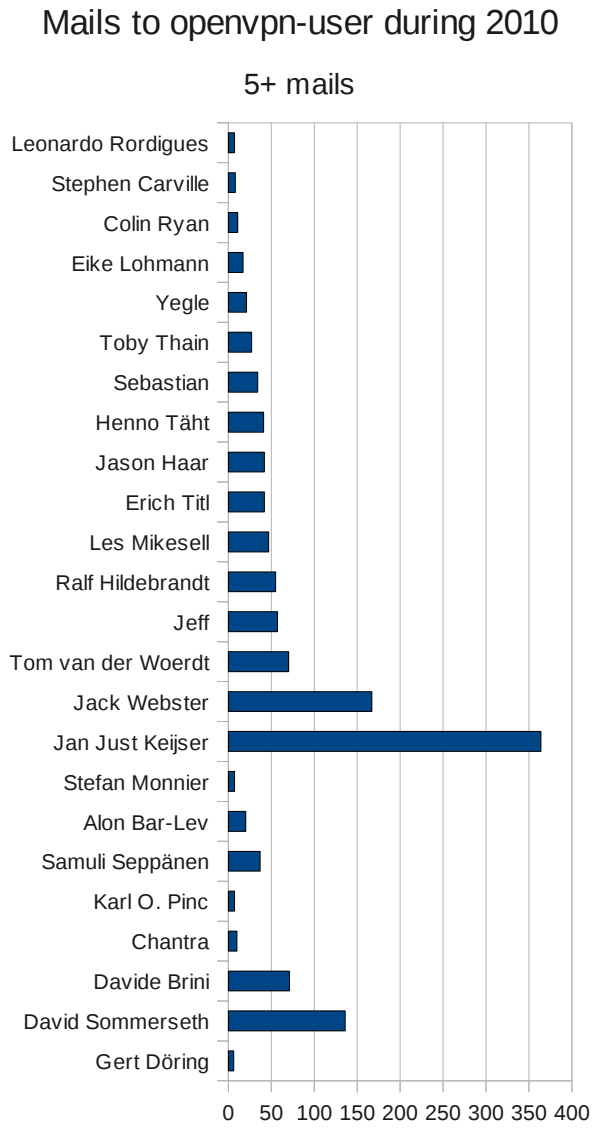

Illustration 6: Mails to openvpn-user during 2010

Alberto Gonzalez Iniesta (1):

Debian patch: Fix spelling in log message

Bernhard Schmidt (2):

Merge remote branch 'jjo/openvpn-testing-feat_ipv6_payload' into gert-ipv6

Merge commit 'remotes/gert/master' into gert-ipv6

Dan Nelson (1):

bash->bourne script cleanup

Daniel Johnson (1):

When I began testing OpenVPN v2.1_rc9 I was having trouble authenticating to the MS Active Directory through auth-pam and Samba. I used the following line in my configs (without the linebreak of course):

David Sommerseth (32):

Add comile time information/settings from ./configure to --version

Added a little bootstrap script for preparing the allmerged branch

Added mapping files from SVN commit ID to more descriptive commit IDs.

Added Python-based build system for Windows in win directory.

Attempt to fix issue where domake-win build system was not properly signing drivers and .exe files.

Avoid repetition of "this config may cache passwords in memory" (v2)

bootstrap.sh - don't use /bin/bash, use /bin/sh instead

Clarified --explicit-exit-notify man page entry

Do not randomize resolving of IP addresses in getaddr()

Don't add compile time information if --enable-small is used

Don't use () in version string - it breaks autotools distcheck

Fix autotools cross-compiling support

Fix dependency checking for configure.h (v2)

Fixed client hang when server don't PUSH (aka the NO_SOUP_FOR_YOU patch)

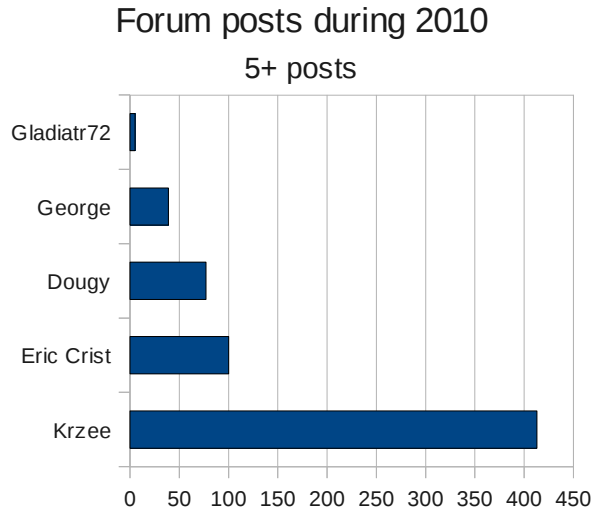

Illustration 7: Forum posts during 2010

Fixed compiler warnings reported on Ubuntu 10.04

Fixed potential NULL pointer issue

Fix multiple configured scripts conflicts issue (version 2)

Harden create_temp_filename() (version 2)

Make use of automake CLEANFILES variable instead of clean-local rule

Make use of counter_type instead of int when counting bytes and network packets

More t_client.sh updates - exit with SKIP when we want to skip

OCSP_check.sh: new check logic

On TARGET_LINUX define _GNU_SOURCE if not defined

Removed no longer needed delete_file() call

Renamed all calls to create_temp_filename()

Revamped the script-security warning logging (version 2)

Reworked the eurephia patch for inclusion to the openvpn-testing tree

Solved hidden merge conflict between feat_misc and bugfix2.1

Test framework improvment - Do not FAIL if t_client.rc is missing

Updated the man page to reflect the behavioural change of create_temp_file()

Use a version which is more understandable by OpenVPN-GUI

verb 5 logging wrongly reports received bytes

Davide Brini (5):

Enhance contrib/pull-resolv-conf/client.{up,down} scripts

Exclude ping and control packets from activity

Fix certificate serial number export

Fix missing /bin/bash -> /bin/sh

The man page does not mention that the default value of "mssfix" is 1450.

Emilien Mantel (2):

Choose a different field in X509 to be username

Fixed static defined length check to use sizeof()

Enrico Scholz (1):

Allow 'lport 0' setup for random port binding

Fabian Knittel (1):

ssl.c: fix use of openvpn_run_script()'s return value

Gert Doering (34):

2.2-beta3 has a signed TAP driver with the IPv6 code, but it's not version 9.7 as anticipated (that's 2.1.3) but 9.8 - change test to require 9.8, and change message to point to 2.2-beta3 and up.

4a, 9, 10, 11, 12 added - and 11. done right away :-) (cherry picked from commit ea382a1d550ac100d27c8118777e3160c85d06d2)

add IPv6 route add / route delete code for windows (using "netsh")

add some TODOs to TODO.IPv6

--version: change printing of IPv6 payload patch version to [...] style

fix "make check" regression in tun.c (unnecessary change reverted)

basic documentation of IPv6 related options and their syntax

bugfix for linux/iproute2: IPv6 ifconfig code block was not called for "dev tun"+"topology subnet"

Build t_client.sh by configure at run-time.

bump IPv6 version number (openvpn --version) to 20100922-1 document Linux tun fixes and test results in ChangeLog.IPv6

correct URL for "more information about IPv6 patch is *here*"

document recent changes and open TODOs, adapt --version info, tag release

drop "book ipv6" from open_tun() and tuncfg() prototypes

Enable IPv6 Payload in OpenVPN p2mp tun server mode. 20100104-1 release.

env_block(): if PATH is not set, add standard PATH setting to env so that "netsh.exe" can find "framedyn.dll" (needs work)

Fix compile problems on NetBSD and OpenBSD

fix date format mistake in PRODUCT_TAP_RELDATE (Peter Stuge)

Fix <net/if.h> compile time problems on OpenBSD for good

Fix problem with special case route targets ('remote_host')

full "VPN client connect" test framework for OpenVPN run from "make check" if "t_client.rc" is found in workdir or srcdir

implement IPv6 ifconfig + route setup/deletion on OpenBSD destroy tunX interface on tun_close() tested on OpenBSD 4.7

Implement IPv6 in TUN mode for Windows TAP driver.

Improved man page entry for script_type

NetBSD fixes - on 4.0 and up, use multi-af mode. On earlier systems that do not have TUNSIFHEAD (and do not have IPv6 capable tunnels), fall back to old IPv4-only code without address-family prepending.

new feature: "ifconfig-ipv6-push" (from ccd/ config)

print patch version on "--version"

remove duplicate code in FREEBSD+DRAGONFLY system-dependent ifconfig

remove NOTES file from commit - private scribbling

renamed t_client.sh to t_client.sh.in build t_client.sh by configure at run-time, with proper paths to ip/ifconfig/netstat binaries, and (most important) with proper #!SHELL

revert unconditionally-enabling of setenv_es() logging (too noisy), replace with #ifdef DEBUG_VERBOSE_SETENV compile-time flag

undo accidential duplication of existing "--iroute" line in the help text

when deleting a route on win32, also add gateway address (otherwise netsh.exe will succeed, but silently ignore request)

WIN32: if IPv6 requested in TUN mode, and TUN/TAP driver version is older than 9.7, log warning and disable IPv6 (won't work anyway).

- Win32 IPv6 ifconfig support, using "netsh" calls

Win32: set next-hop for IPv6 routes according to TUN/TAP mode - in TUN mode, use special next-hop address (fe80::8) that tapdrv will handle ND for

James Yonan (41):

Added "net stop dnscache" and "net start dnscache" in front of existing --register-dns commands.

Added --proto-force directive.

Added Python-based build system for Windows in win directory.

Added --register-dns option for Windows.

Added stub directive "remote-ip-hint".

Added support for MSVC debugging of openvpn.exe in settings.in:

Added warning about tls-remote in man page.

Added win/build_exe.py script, which is similar to win/build_all.py except that it doesn't build the TAP drivers or tapinstall.

Allow PKCS12 file content to be included inline in configuration file, rendered as base64.

Attempt to fix issue where domake-win build system was not properly signing drivers and .exe files.

Distribute win directory (Python/MSVC-based build system) in "make dist" tarball.

Don't advance the connection list on AUTH_FAILED errors.

Don't configure Linux tun/tap txqueuelen setting if OpenVPN txqueuelen directive is set to 0.

Fixed an issue in the Management Interface that could cause a process hang with 100% CPU utilization in --management-client mode if the management interface client disconnected at the point where credentials are queried.

Fixed an issue where application payload transmissions on the TLS control channel (such as AUTH_FAILED) that occur during or immediately after a TLS renegotiation might be dropped.

Fixed an issue where AUTH_FAILED was not being properly delivered to the client when a bad password is given for mid-session reauth.

Fixed an issue where if reneg-sec was set to 0 on the client, so that the server-side value would take precedence, the auth_deferred_expire_window function would incorrectly return a window period of 0 seconds. In this case, the correct window period should be the handshake window period.

Fixed bug in proxy fallback capability where openvpn.exe could core dump if http-proxy-fallback-disable command was issued in response to ">PROXY:NEED_NOW management" interface notification.

Fixed compiling issues when using --disable-crypto

Fixed initialization bug in route_list_add_default_gateway (Gert Doering).

Fixed issue on Windows with MSVC compiler, where TCP_NODELAY support was not being compiled in.

Fixed issue where bad creds provided by the management interface for HTTP Proxy Basic Authentication would go into an infinite retry-fail loop instead of requerying the management interface for new creds.

Fixed typo: missing comment close.

Fixes to prevent compile breakage when --disable-crypto is used.

Implement challenge/response authentication support in client mode, where credentials are entered

from stdin. This capability is compiled when ENABLE_CLIENT_CR is defined in syshead.h (enabled by default).

Implemented a key/value auth channel from client to server.

Implemented http-proxy-override and http-proxy-fallback directives to make it easier for OpenVPN client UIs to start a pre-existing client config file with proxy options, or to adaptively fall back to a proxy connection if a direct connection fails.

Implemented multi-address DNS expansion on the network field of route commands.

In verify_callback, the subject var should be freed by OPENSSL_free, not free, since it is allocated by OpenSSL.

Make base64.h have the same conditional compilation expression as base64.c.

Management interface performance optimizations:

Minor change to doclean script:

Minor fixes to recent HTTP proxy changes:

Modified ">PASSWORD:Verification Failed" management interface notification to include a client reason string:

Proxy improvements:

Set socket buffers (SO_SNDBUF and SO_RCVBUF) immediately after socket is created rather than waiting until after connect/listen.

Trivial fix to proxy.c -- #define proxy auth type as UP_TYPE_PROXY.

Updated build scripts to work with the IPv6 enabled Windows TAP driver

Updated copyright date to 2010.

Updated MSVC build scripts to Visual Studio 2008: python msvc\config.py nmake /f msvc\msvc.mak

When aborting in a non-graceful way, try to execute do_close_tun in init.c prior to daemon exit to ensure that the tun/tap interface is closed and any added routes are deleted.

Windows security issue: Fixed potential local privilege escalation vulnerability in Windows service.

Jan Brinkmann (1):

The man page needs dash escaping in UTF-8 environments

Jesse Young (1):

Remove hardcoded path to resolvconf

JuanJo Ciarlante (31):

* added README.ipv6.txt

* correctly setup hints.ai_socktype for getaddrinfo(), althought sorta hacky, see TODO.ipv6.

* created getaddr6(), use it from resolve_remote() next: merge ipv{4,6} signal logic into one inside resolve_remote() * passes {loopback,remote}{udp,tcp}{4,6} tests

* document ipv6 milestone status

* doc updates

* doc update w/unittests results

* fix --disable-ipv6 build

* fixed segfault for undef address family in print_sockaddr_ex (thanks Marcel!)

* fixed win32 non-ipv6 build

* fix --multihome for ipv4: cmsg_len must compare against in_pktinfo size, not the full 4+6 union, also use saner variable names.

* fix multi-tcp crash (corrected assertion)

* important fix for tcp6 reconnection was incorrectly creating a PF_INET socket

* init.c: document the ENABLE_MANAGEMENT place to work on

* init.c: small in-doc tweaks

* ipv6 on win32 "milestone": 1st snapshot that passes all unittests

* make ipv6_payload compile under windowze - create inet_ntop() and inet_pton() wrap-implementations using WSAAddressToString() and WSAStringToAddress() functions - add relevant win32-only headers to syshead.h NOTE: syshead.h changes are already included in ipv6_transport

* make possible to x-compile openvpn/win32 in Linux

* migrated all getaddrinfo() to getaddr6 * tests Ok: {loopback,remote}{udp,tcp}{4,6}

* no new funcionality, just small cleanups: - cmdline options help: add tcp6/udp6 missing messages - win32: expand usage of proto_is_udp(), proto_is_tcp() - replace some memset(&obj, 0, sizeof obj) by openvpn's CLEAR(obj)

* openbsd: no IFF_MULTICAST, #ifdef around it

* polished redirect-gateway (ipv4 on ipv6 endpoints) support

* (prototype) fix for supporting "redirect-gateway" for tunneled ipv4 over ipv6 endpoints

* rebased openvpn-2.1_rc1b.jjo.20061206.d.patch * passes {udp,tcp}x{v4,v6} loopback tests * passes {udp,tcp}x{v6} remote tests

* rebased to v2.1.1 release * document {un,}trusted_ip6 in manpage

* renamed README.ipv6{.txt,}

* socket.c: better buf logic in print_sockaddr_ex

* socket.c: use USE_PF_INET6 in switch constructs to actually toss them out, GNU indentation for my deltas

* support --disable-ipv6 build properly: - tests now are pass (and fail) properly for ipv6/4 builds *

more GNU indenting

* TODO.ipv6 update
* undo mroute.c changes related to ipv6 payload, nothing to do w/ipv6 transport afterall.
* updated doc
* updated {README,TODO}.ipv6 from feedback at openvpn-devel mlist

Karl O. Pinc (2):

Change verify-cn so cn is no longer hardcoded in openvpn's config file

Several updates to openvpn.8 (man page updates)

Lars Hupel (1):

Add HTTP/1.1 Host header

Mathieu GIANNECCHINI (1):

enhance tls-verify possibility

Samuli Seppänen (1):

Added command-line option parser and an unsigned build option to build_all.py

Wil Cooley (1):

pkitool lacks expected option "--help"

chantra (3):

Handle non standard subnets in PF grammar

Fix errors in openvpn-plugin.h documentation

Fixes openssl-1.0.0 compilation warning