Proxytunnel

Punching holes through
the corporate firewall.

Mark Janssen – Dag Wieërs
maniac@maniac.nl – dag@wieers.com

Proxytunnel history

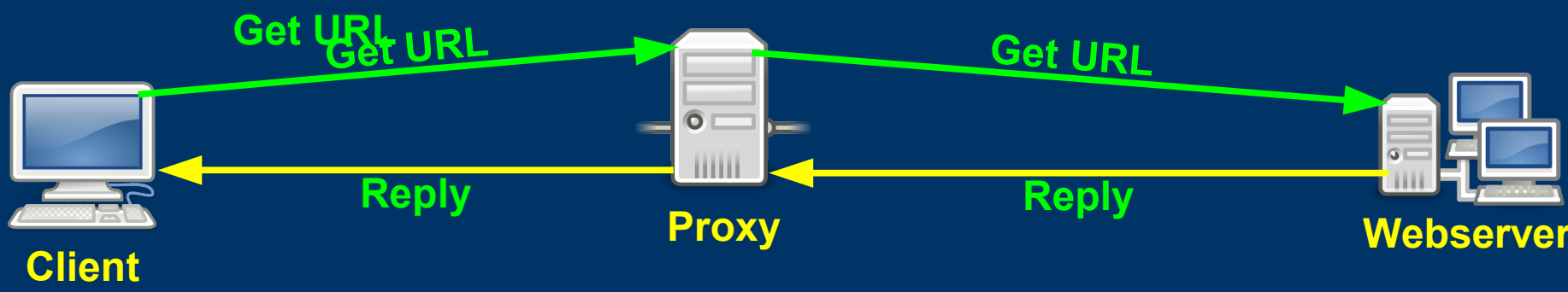
- Being stuck in a corporate network back in 2001
- Jos Visser and Mark came up with the Proxytunnel idea and wrote the first implementation.
- First cvs import in SourceForge in November 2001 (older history not recorded).
- Features: basic authentication, and getting through standard web proxies
- Very small codebase (2 .c, 3 .h, Makefile)

Growing...

- During the following years various new features were added as the need arose:
  - Multiple platforms supported (unix, os-x, windows)
  - Sending extra headers
  - NTLM Authentication
  - Proxy Bouncing
  - SSL wrapping
- Only possible due to help from the community!
  - Dag Wieërs, Fred Donck, Paul Solomon, Alex Peuchert, Mark Cave-Ayland, and many others...

How a typical web proxy works

1. Browser connects to proxy
2. Browser requests URL
3. Proxy connects to webserver and sends request
4. Webserver responds to the proxy
5. Proxy copies data back to the browser



Just so you know

- The proxy server can allow or deny requests based on local policy.
- Nothing we can do about that.
- The proxy can require authentication before use
- The proxy can see all traffic going through it, and can even modify it in transit.

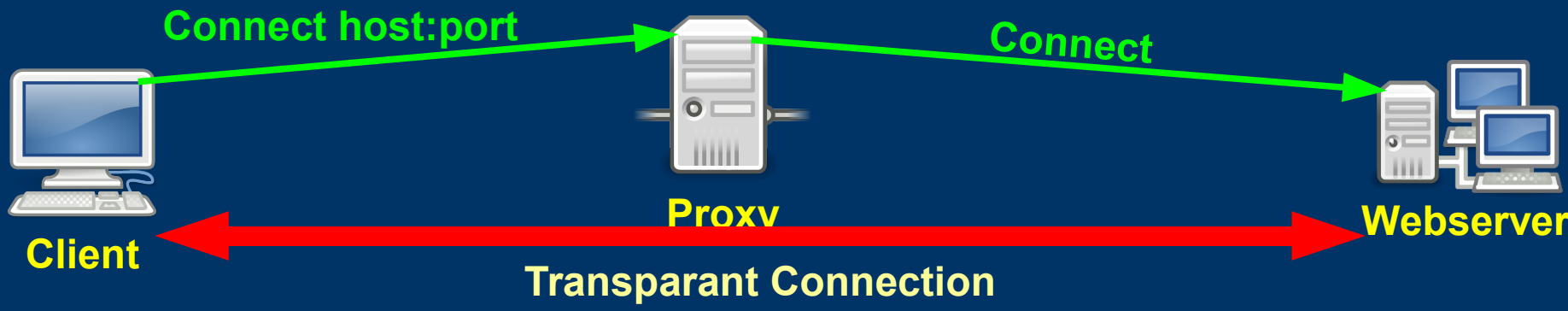
- This is NOT the case on HTTPS requests (at least, not without us knowing)

Web proxy with https requests

- With SSL traffic, the webbrowser and proxy interact differently.
- The browser connects to the proxy, and asks it to create a connection to the webserver, forwarding traffic between browser and webserver
- The browser then negotiates the encryption-protocol and keys with the webserver and finally makes a request for a URI
- The proxy has no idea what the browser is doing

Http CONNECT over a proxy

- The proxy creates a transparant connection for the browser and doesn't interfere.



What can we do

- If we can browse to arbitrary secure sites (try your banking website, GMail, etc)
- Then we can most likely connect to our own controlled secure site
- We can get the proxy server to create a direct connection for us using http's CONNECT method.
- ProxyCommand in OpenSSH can be used to send the CONNECT to your proxy.

How to get OpenSSH to connect

- If there are no restrictions
  - Just run ssh normally, using port 22
- If there is a transparant proxy, or a port-limit
  - Run sshd on port 443
- If there is a non-transparant proxy
  - Use netcat to send the connect

```
netcat -X connect -x proxy:port <host> <port>
```

- If the proxy uses authentication, protocol-inspection or you want to do more:
  - Use Proxytunnel

Proxytunnel takes care of...

- Connecting to the proxy
- Authenticating with basic or NTLM authentication as needed
- Asking the proxy to make the connection
- Optionally do some magic
- Forward traffic from the user over the connection and vice-versa

Typical use-case

- Run proxytunnel as a ProxyCommand in OpenSSH
- Have sshd(8) listening on port 443 of a controlled system.
- Use ssh(1) as normally, using ssh's portforwarding and socks-capabilities to get an unfiltered and encrypted connection to a trusted/ controlled system.

Ssh(1) configuration

- ~/.ssh/config

```
Host shell.home.net
   ProxyCommand proxytunnel [options]↵
     -p <proxy>:<port> -d %h:443
   DynamicForward 1080
   ServerAliveInterval 20
   ServerAliveCountMax 5
```

Methods to give auth passwords

- On the commandline → easy, quick'n'dirty
- In env variable → findable in /proc (by root)
- In a file → not safe against root, otherwise ok
- Have proxytunnel prompt for it
  - Most secure/safe, but interactive

- In future, maybe use keymanager ?

Demo

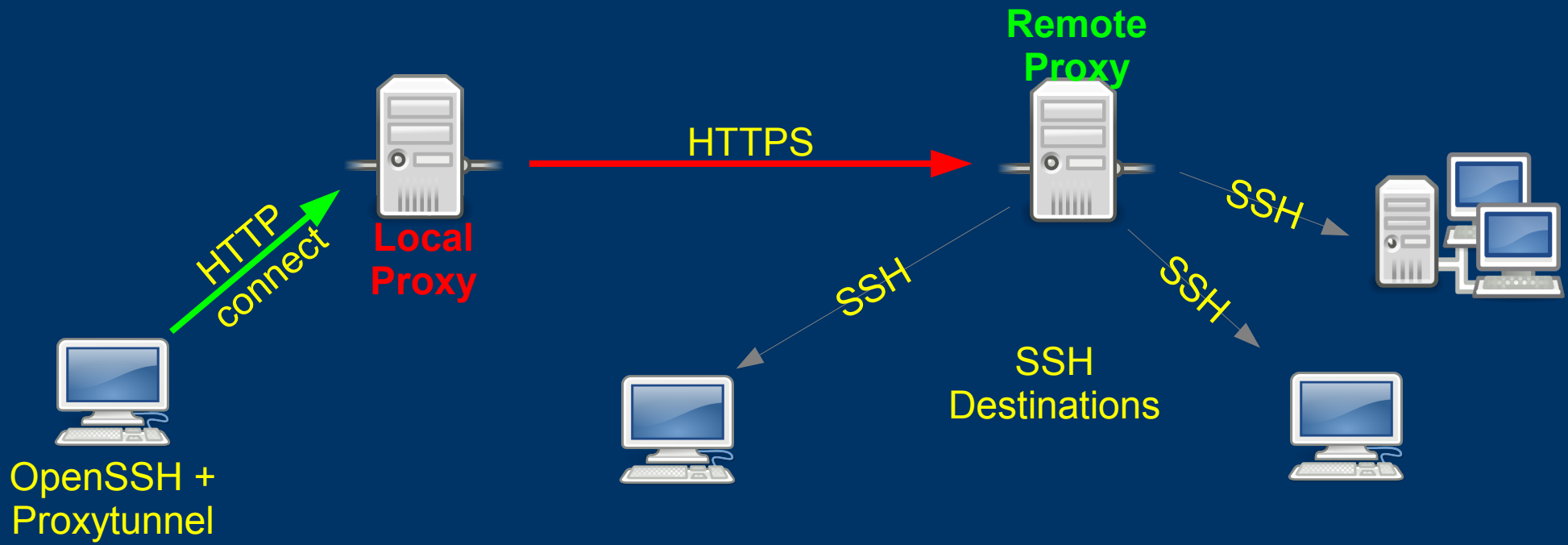
- Basic connect, no authentication
- Connect with basic http-authentication
- Connect using SSL to local proxy
- Connect to proxy, use SSL to endpoint

Advanced use...

- Authentication (basic + ntlm)
  - Windows networks, IIS proxies
- Additional headers
  - For stealth, or because proxy requires it
  - Method to support extensions/testing
- Evading deep-packet-inspection / protocol inspection
  - Look more like regular https traffic, by using SSL
  - Requires SSL support on server (stunnel4)
- Setproctitle
  - Borrowed from openssh-portable
  - Useful for process-hiding on shared systems

Proxy bouncing

- We only have the HTTP CONNECT method
- But if we own another proxy/apache we can do anything we want, instructed by OpenSSH



Proxy bouncing features

- Much more flexibility to connect anywhere
  - destination is resolved on your proxy
- Offers normal web pages to mask tunneling
  - effectively hides it for security people
- Does not need any special software
  - apache is a very secure and trusted project
- Works for any situation, even the simple ones

BUT Apache does not allow (by default) to use CONNECT over SSL using mod_connect

  - There is a patch at bug #29744, please help us shout

Proxy bouncing configuration

- Apache mod_connect directives

```
ProxyRequests on
AllowConnect 22 2022
```

- Apache access control (for source/destination)

```
<Proxy *>
        Order deny,allow
        Deny from all
</Proxy>
<ProxyMatch “^(wieers.com|.+\.rpmforge.net):”>
        Order deny,allow
        Allow from proxy.customer.com
</ProxyMatch>
```

Proxy bouncing authentication

- Apache authentication (easier and more useful)

```
<Proxy *>
    Order allow,deny
    Allow from all
    AuthType Basic
    AuthName “Some string”
    AuthUserFile /some/path/htpasswd
    Require valid-user
</Proxy>
```

Demo continued

- Connect with proxy bouncing
- Proxy bouncing with authentication on remote
- Proxy bouncing with SSL between local and remote proxy

Proxytunnel compared

- Corkscrew
  - Basic functions only, basic-auth since 2.0
- GNU httptunnel / HTun
  - Works over http, requires server-component
- PrTunnel
  - Basic functions, untested basic-auth, does SOCKS
- SOHT
  - Works over http, java-based server component

Open issues

- Apache mod_proxy will not accept CONNECT requests over SSL
  - Politics, won't-fix, patch+workarounds exist #29744
  - Possible to use stunnel4 as workaround
- Setproctitle doesn't work on all platforms
- Windows build doesn't handle debug-info correctly

- Todo: read settings from .proxytunnelrc

Thank you for listening

Any questions ?

http://proxytunnel.sourceforge.net/
http://dag.wieers.com/howto/ssh-http-tunneling/