TrustBase

Repairing and Strengthening Certificate-based Authentication

certificate validation problems
certificate authorities (CAs)

- generally can sign certificates for any host (Eckersley et al.)
- have been hacked, sometimes repeatedly (Marlinspike)
- can be influenced and operated by governments (Soghoian et al.)
- don’t always follow best practices (see CNNIC)
for application developers

• mobile and desktop apps have validation problems
  • Brubaker et al., Georgiev et al., Onwuzurike et al., Fahl et al.
• security libraries are complicated
• security may not be a priority
threat model
alternate and reinforcing strategies

• mitigate many of these issues
• have no common platform or API
• have difficulty being adopted
trust decisions are outsourced
how do we enable admins to control trust decisions for their own machines?
TrustBase

• motivating principles
  • centralize authentication as an OS service
  • empower system admins to dictate how trust decisions are made

• design goals
  • secure all existing applications
  • prohibit unprivileged applications from acting against administrator rules
  • provide easy deployment of authentication systems
  • negligible overhead
moving trust to the OS

Prototypes for
• Linux
• Android (nonrooted)
• Windows
TrustBase architecture
TrustBase architecture
traffic interception (Linux)

• loadable kernel module
• hooks into native transport protocol functionality
• provides generic inspection/modification API
TrustBase architecture
TLS handler

1. monitor traffic for TLS records
2. record handshake messages
3. query policy engine with handshake data
4. receive policy response
   1. block connection if invalid
   2. allow if valid

stop tracking TLS/SSL?

full handshake?

start tracking

yes

no
TrustBase architecture

Plugin API
- Pinning
- Notaries
- Revocation
- New Services

Policy Engine
- Configuration
- Notifications
- Handlers
- Validation API
- Traffic Interceptor
- Modified Application
- Unmodified Application
policy engine

• receives queries via Netlink
• implements basic CA validation
• aggregates decisions from plugins
  • necessary
  • voting
• provides native API
  • Linux capabilities
TrustBase architecture
plugins

- API allows synchronous and asynchronous plugins
  - openssl STACK_OF(X509) or ASN.1 DER
- can report back yes/no/abstain/error for each chain
- have access to all handshake info (and more)

addons

- provide additional language support for plugins
- currently have native C and python addons
- API to add additional language support
included plugins and uses

• CA validation (builtin)
• certificate pinning
• OSCP checking
• CRLSet blocking
• DANE
• notary
• cipher suite auditor
evaluation
centralization and coverage

con
• single point of failure

pro
• updates are global
• benefit of many eyes

• TrustBase makes connection security an OS service, like TCP, IP

<table>
<thead>
<tr>
<th>Library</th>
<th>Tool</th>
</tr>
</thead>
<tbody>
<tr>
<td>C++</td>
<td>gnutls-cli</td>
</tr>
<tr>
<td>libcurl</td>
<td>curl</td>
</tr>
<tr>
<td>libgnutls</td>
<td>ssllscan</td>
</tr>
<tr>
<td>libssl</td>
<td>openssl s_client</td>
</tr>
<tr>
<td>libnss</td>
<td>openssl s_time</td>
</tr>
<tr>
<td>JAVA</td>
<td>lynx</td>
</tr>
<tr>
<td>SSLSocketFactory</td>
<td>fetchmail</td>
</tr>
<tr>
<td>PERL</td>
<td>firefox</td>
</tr>
<tr>
<td>socket::ssl</td>
<td>chrome/chromium</td>
</tr>
<tr>
<td>PHP</td>
<td>mpop</td>
</tr>
<tr>
<td>fsockopen</td>
<td>w3m</td>
</tr>
<tr>
<td>php_curl</td>
<td>ncat</td>
</tr>
<tr>
<td>PYTHON</td>
<td>wget</td>
</tr>
<tr>
<td>httplib</td>
<td>steam</td>
</tr>
<tr>
<td>httplib2</td>
<td>thunderbird</td>
</tr>
<tr>
<td>pycurl</td>
<td>kmail</td>
</tr>
<tr>
<td>pyOpenSSL</td>
<td>pidgin</td>
</tr>
<tr>
<td>python ssl</td>
<td></td>
</tr>
<tr>
<td>urllib, urllib2, urllib3</td>
<td></td>
</tr>
<tr>
<td>requests</td>
<td></td>
</tr>
</tbody>
</table>
hardening

• unprivileged malware cannot unload interception
• CAP_NET_RAW is required to use raw sockets (default) and to bypass TrustBase interception
• CAP_NET_ADMIN required to receive and respond to queries
• configuration is writable only by privileged users
• daemons run nonroot with only required permissions
performance
future work

• POSIX-based secure socket API
• push all of TLS to admin/OS control
• ease developer burden further
• call TrustBase validation natively
• wouldn’t this be nice?

    int socket = socket(PF_INET, SOCK_STREAM, IPPROTO_TLS);
trustbase lets you trust who you want how you want
Thank You

• source code: https://github.com/markoneill/trustbase-linux

• pull requests welcome!
• project website: https://owntrust.org
• contact me: mto@byu.edu

• thanks to our sponsors: