AbUsing TLS for Defensive Wins

Lee Brotherston - @synackpse - #TLSAbuse
SSL TLS does what now?

Lee Brotherston - @synackpse - #TLSAbuse
Current State of the Art
- Ignore TLS
- Break TLS
- Embrace TLS

Lee Brotherston - @synackpse - #TLSAbuse
Ignore TLS
## IDS Rules

<table>
<thead>
<tr>
<th>Protocol</th>
<th>ClearText</th>
<th>TLS Enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP(S)</td>
<td>1572</td>
<td>25</td>
</tr>
<tr>
<td>IMAP(S)</td>
<td>34</td>
<td>10</td>
</tr>
<tr>
<td>SMTP(S)</td>
<td>73</td>
<td>10</td>
</tr>
</tbody>
</table>

Lee Brotherston – @synackpse – #TLSAbuse
permit any any port 443

Universal Firewall Bypass Port!

Lee Brotherston - @synackpse - #TLSAbuse
ssh -p443 user@myhost

(don’t pretend you don’t)
So, what then?
End MalVirus Point Ransom WhitelisterWare Protection
Blacklists

Lee Brotherston - @synackpse - #TLSAbuse
Break TLS

Lee Brotherston - @synackpse - #TLSAbuse
Lee Brotherston - @synackpse - #TLSAbuse
Break The CA Model

.... even more

Lee Brotherston - @synackpse - #TLSAbuse
Compromised Appliance

Lee Brotherston - @synackpse - #TLSAbuse
Compromised

Internal CA/Key

Lee Brotherston - @synackpse - #TLSAbuse
Key Management is Hard

Lee Brotherston - @synackpse - #TLSAbuse
Poor Certificate Validation

Lee Brotherston - @synackpse - #TLSAbuse
“Trusted” CA List

Lee Brotherston - @synackpse - #TLSAbuse
Trust What The Appliance Trusts

*cough* WoSign *cough*

Lee Brotherston - @synackpse - #TLSAbuse
Certificate

[un]Pinning

Lee Brotherston - @synackpse - #TLSAbuse
Malicious Insider

Lee Brotherston - @synackpse - #TLSAbuse
Embracing TLS?

Lee Brotherston - @synackpse - #TLSAbuse
Goals

Lee Brotherston - @synackpse - #TLSAbuse
“Perfection is the enemy of good enough”

Lee Brotherston - @synackpse - #TLSAbuse
Lee Brotherston - @synackpse - #TLSAbuse
TLS Fingerprinting

(I hear someone did a talk on that)

Lee Brotherston - @synackpse - #TLSAbuse
Spotting $bad

Lee Brotherston - @synackpse - #TLSAbuse
Introduction

TLS (Transport Layer Security) is a cryptographic protocol that provides privacy for applications. TLS is usually implemented on top of common protocols such as HTTP for web browsing or SMTP for email. HTTPS is the usage of TLS over HTTP, which is the most popular way of securing communication between a web server and client and is supported by the bulk of major web servers.

As TLS has become more popular and easier to use, we have seen the adoption of this technology by malware to secure its own communication. It is fairly straightforward for malware to plug into existing TLS libraries, and in some cases include an entire implementation in its own source code. This ease of use is troubling because it allows malware to easily evade detection and blend into benign traffic patterns typically observed on a network. In short, malware authors know how to use encryption, and they use it in TLS and in
Pseudo “anomaly based detection”

Lee Brotherston - @synackpse - #TLSAbuse
Man-in-the-Middle Interfering with Increased Security

According to the plan we published earlier for deprecating SHA-1, on January 1, 2016, Firefox 43 began rejecting new certificates signed with the SHA-1 digest algorithm. For Firefox users with unfiltered access to the Internet, this change probably went unnoticed, since there simply aren’t that many new SHA-1 certs being used. However, for Firefox users who are behind certain “man-in-the-middle” devices (including some security scanners and antivirus products),
Incident Response & Attribution (ish)

Lee Brotherston - @synackpse - #TLSAbuse
Vorführeffekt
OK, Enough Fingerprinting
Server Responses

Lee Brotherston - @synackpse - #TLSAbuse
The Superfish certificate has been cracked, exposing Lenovo users to attack

By Russell Brandon on February 19, 2014 10:45 am  Email  @russellbrandt
Do You Even IDS, Bro?

Lee Brotherston - @synackpse - #TLSAbuse
Inline TLS Shenanigans

Lee Brotherston - @synackpse - #TLSAbuse
TLS Handshake Mangling

Lee Brotherston - @synackpse - #TLSAbuse
Nope
Subtractive only

Lee Brotherston - @synackpse - #TLSAbuse
Appendix A: Additional PCI DSS Requirements for Enabling SSL and TLS

Enabling SSL and/or TLS requires an organization to assess the potential security risks associated with enabling these protocols and to implement controls to mitigate those risks. The PCI DSS requirements are designed to help organizations assess and implement appropriate controls to mitigate these risks.

The PCI DSS requirements do not require SSL and/or TLS to be enabled or disabled. Instead, they require organizations to assess the potential security risks associated with enabling these protocols and to implement controls to mitigate those risks.

Recommendations for Enabling SSL and TLS

- Organizations should consider the potential security risks associated with enabling SSL and TLS.
- Organizations should implement controls to mitigate those risks.
- Organizations should regularly review and update their SSL and TLS configurations.

These recommendations are based on the PCI DSS requirements and best practices for enabling SSL and TLS.

Lee Brotherston - @synackpse - #TLSAbuse
The Remaining Problem
This site can't provide a secure connection

www.squarelemon.com sent an invalid response.

Try running Network Diagnostics.

ERR_SSL_PROTOCOL_ERROR

Reload
Secure Connection Failed

An error occurred during a connection to www.google.com. SSL received an unexpected Server Hello Done handshake message. Error code: SSL_ERROR_RX_UNEXPECTED_HELLO_DONE

- The page you are trying to view cannot be shown because the authenticity of the received data could not be verified.
- Please contact the website owners to inform them of this problem.

Learn more...

Try Again

Report errors like this to help Mozilla identify and block malicious sites
A Solution.....

Kinda

Lee Brotherston - @synackpse - #TLSAbuse
This site can’t be reached

demo.squarelemon.com refused to connect.

Try:
- Checking the connection
- Checking the proxy and the firewall

ERR_CONNECTION_REFUSED

DETAILS

Reload
Unable to connect

Firefox can’t establish a connection to the server at demo.squarelemon.com.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer’s network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the Web.

Try Again
How Did we Do?

- Coarse Content Filtering ✓
- Application Detection ✓
- Coarse Logging ✓
- Granular Logging …?
- Certificate Checks ✓
- Malware Detection ✓
- In-Page Exploits ✗
- Data Exfiltration ✓
- DLP ✓

Lee Brotherston - @synackpse - #TLSAbuse
GET /XXXXXXXXXXX HTTP/1.1
Host: www.myhost.com
User-Agent: MyBrowser/10.4 (Some OS) CoolWebKit/537.36
Accept: text/html,application/xhtml+xml,application/xml
Accept-Encoding: gzip, deflate, sdch
Accept-Language: en-US,en;q=0.8
Conclusion
Things...

TLS Fingerprinting: https://github.com/LeeBrotherston/tls-fingerprinting
TLS Fingerprinting: https://blog.squarelemon.com/tls-fingerprinting/
TLS Mangler (soon): https://github.com/LeeBrotherston/<somewhere>
TLS Profiling: https://gist.github.com/wxsBSD/6d5e777afc31b3cf46d0
TLS Profiling: https://gist.github.com/wxsBSD/0c6584913bcc5e6da31b
Slide Deck: https://speakerdeck.com/leebrotherston/<staytuned>
The Twitters (me): @synackpse
Any Questions?