sslstrip,
and the issues we encountered

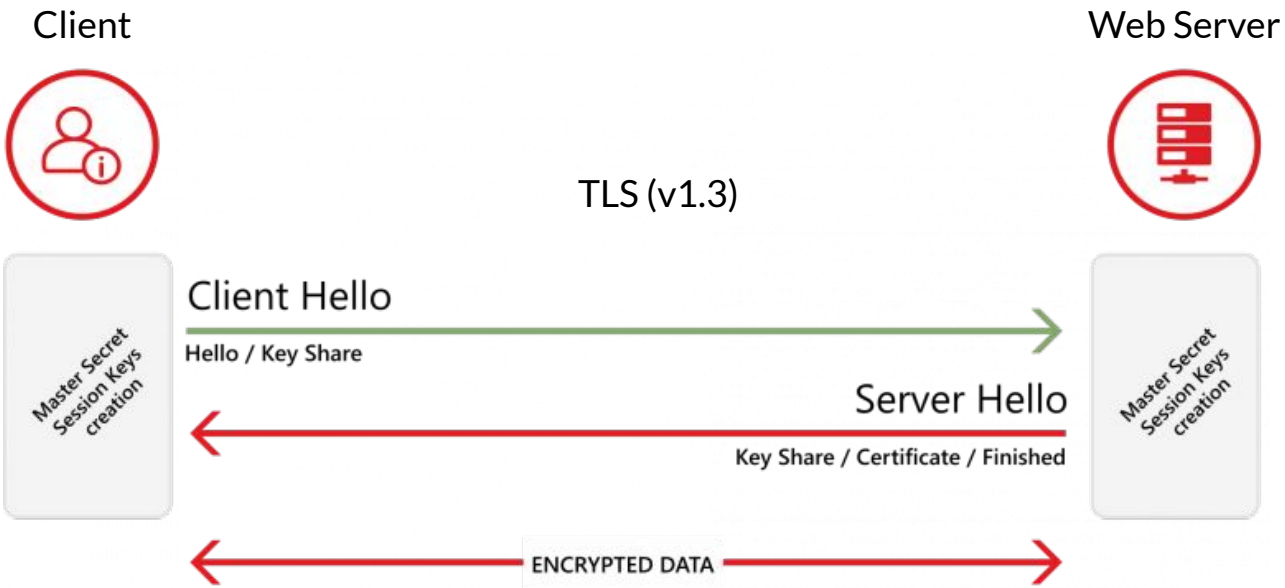
Software Composition Seminar

22 December 2020

Alex Nyffenegger

How does *HTTPS* work?

HTTP uses plain text and is secured by the *Transport Layer Security (TLS)* or its predecessor *Secure Sockets Layer (SSL).*



https://securityboulevard.com/2019/03/tls-security-5-establishing-a-tls-connection/

What is *HTTPS stripping*?

2009: **Moxie Marlinspike presents SSL stripping**

Step 1: Replace all HTTPS links in doc
Step 2: Remove security flags from server packets
Step 3: Hide redirects to HTTPS

https://www.semurity.com/quick-guide-to-ssl-tls-hijacking-using-sslstrip/

! stripping ≠ downgrading TLS

stripping ≠ removing TLS

stripping ≠ cracking TLS

Is *HTTP Strict Transport Security* the solution?

2012: HTTP Strict Transport Security (HSTS) standard has been published

HSTS stores visited protected websites for given time and allows preload lists.

But unfortunately, …

… it is not widely available in HTTP clients.

… it is not considered necessary by some developers.

Support HSTS #3170

New issue

Open **yschimke** opened this issue on 19 Feb 2017 · 13 comments

yschimke commented on 19 Feb 2017 — Collaborator

Feature Request: https://tools.ietf.org/html/rfc6797

HSTS with Preloaded site list

Mainly curious if this is something that would fit within OkHttp core, or should be a purely separate addon? To me this seems like something that should be part of the core, because its required to correctly follow the directions of sites that provide the header.

I was experimenting here https://github.com/square/okhttp/compare/master...yschimke:hsts?expand=1

It seems like it should be possible to implement cleanly either way as an application interceptor. Implementing in OkHttp core would ideally built of some persistent configuration support.

swankjesse commented on 18 Mar 2017 — Member

Persistent configuration would be really nice for stuff like this. See also:
#2890

1

swankjesse added the enhancement label on 18 Mar 2017

swankjesse added this to the **3.8** milestone on 18 Mar 2017

swankjesse modified the milestones: **3.10**, **3.11** on 18 Feb 2018

amirlivneh commented on 20 Aug 2018 — Collaborator

We have an implementation of an Interceptor implementing HSTS pre-loaded list (no Strict-Transport-Security header support). I can submit a PR if you decide this should be part of the OkHttp core.

yschimke commented on 23 Aug 2018 — Collaborator Author

@amirlivneh I'd love to integrate a java/kotlin version in my own project if not. Care to publish this somewhere?

Assignees
No one assigned

Labels
enhancement

Milestone
Backlog

Linked pull requests
Successfully merging a pull request may close this issue.
None yet

Notifications Customize
Subscribe
You're not receiving notifications from this thread.

3 participants

amirlivneh commented on Aug 27, 2018 — Collaborator

I agree that disabling cleartext is the simplest solution and probably good enough for most clients. Our apps have special use cases that require more granular control. Fortunately, Interceptors are powerful enough, so we can rely on one to address our needs.

In our solution, we decided not to couple the enforcing mechanism with a public preload list. We have lightweight apps with extreme APK size constraints. Some of these apps only send requests to a limited set of domains. Decoupling the public preload list from the enforcing mechanism allows those apps to benefit from HTTPS forcing without taking the size penalty of a full public list.

👍 2

yschimke commented on May 9 — Collaborator — Author

This seems pretty widely adopted https://caniuse.com/#search=hsts

yschimke commented on Oct 26 — Collaborator — Author

Also relevant https://emilymstark.com/2020/10/24/strict-transport-security-vs-https-resource-records-the-showdown.html

and

https://twitter.com/estark37/status/1320442640123323464

yschimke commented on Oct 26 — Collaborator — Author

https://twitter.com/tunetheweb/status/1320661941556252672

Discusses caching of similar items.

yschimke commented 16 days ago — Collaborator — Author

Curl implementation https://daniel.haxx.se/blog/2020/11/03/hsts-your-curl/
https://github.com/curl/curl/blob/21464a65c60092e410bb3c8195f3160fc49f4286/docs/HSTS.md

okHttp

OkHttp

OkHttp is an efficient HTTP client providing features such as SPDY, connection pooling, transparent compression, and response caching.

Number of apps	**Over 40 Thousand**
Total number of downloads	**Over 43 Billion**
License	Apache License 2.0
Tags	#4 in Network, Open Source
Website	http://square.github.io/okhttp/
GitHub	Watchers: **1.7k**, Stars: **39k**, Forks: **8.3k**

↓ Read more about our statistics

https://www.appbrain.com/stats/libraries/details/okhttp/okhttp

Strict Transport Security - OTHER

Usage % of all users ?

Global 97.79%

Declare that a website is only accessible over a secure connection (HTTPS).

Current aligned | Usage relative | Date relative | Filtered | All

IE	Edge	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	Opera Mobile	Chrome for Android	Firefox for Android	UC Browser for Android	Samsung Internet	QQ Browser	Baidu Browser	KaiOS Browser
		2-3.6		3.1-6.1	10-11.5	3.2-6.1		2.1-4.3								
6-10	12-86	4-83	4-86	7-13.1	12.1-71	7-13.7		4.4-4.4.4	12-12.1				4-12.0			
1 11	87	84	87	14	72	14.2	all	81	59	87	83	12.12	13.0	10.4	7.12	2.5
		85-86	88-90	TP												

Notes | Test on a real browser NEW | Known issues (0) | Resources (6) | Feedback

The HTTP header is 'Strict-Transport-Security'.

1 IE 11 added support in an update on June 9, 2015

Scope of this project

Can we attack

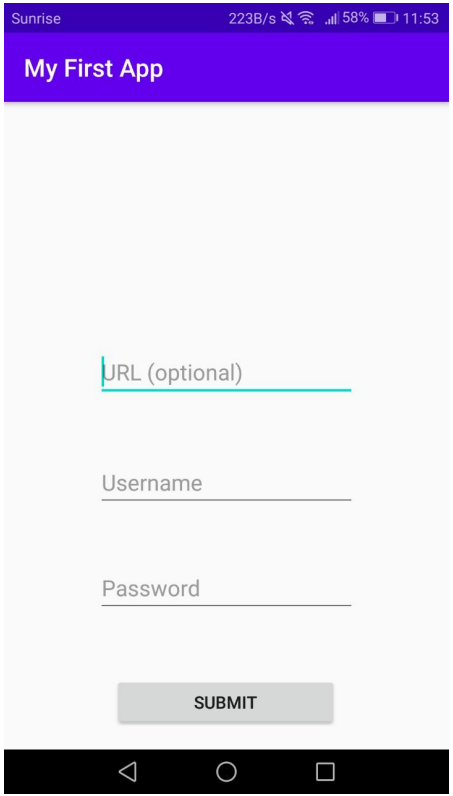
modern Android networking libraries

with the tool sslstripping?

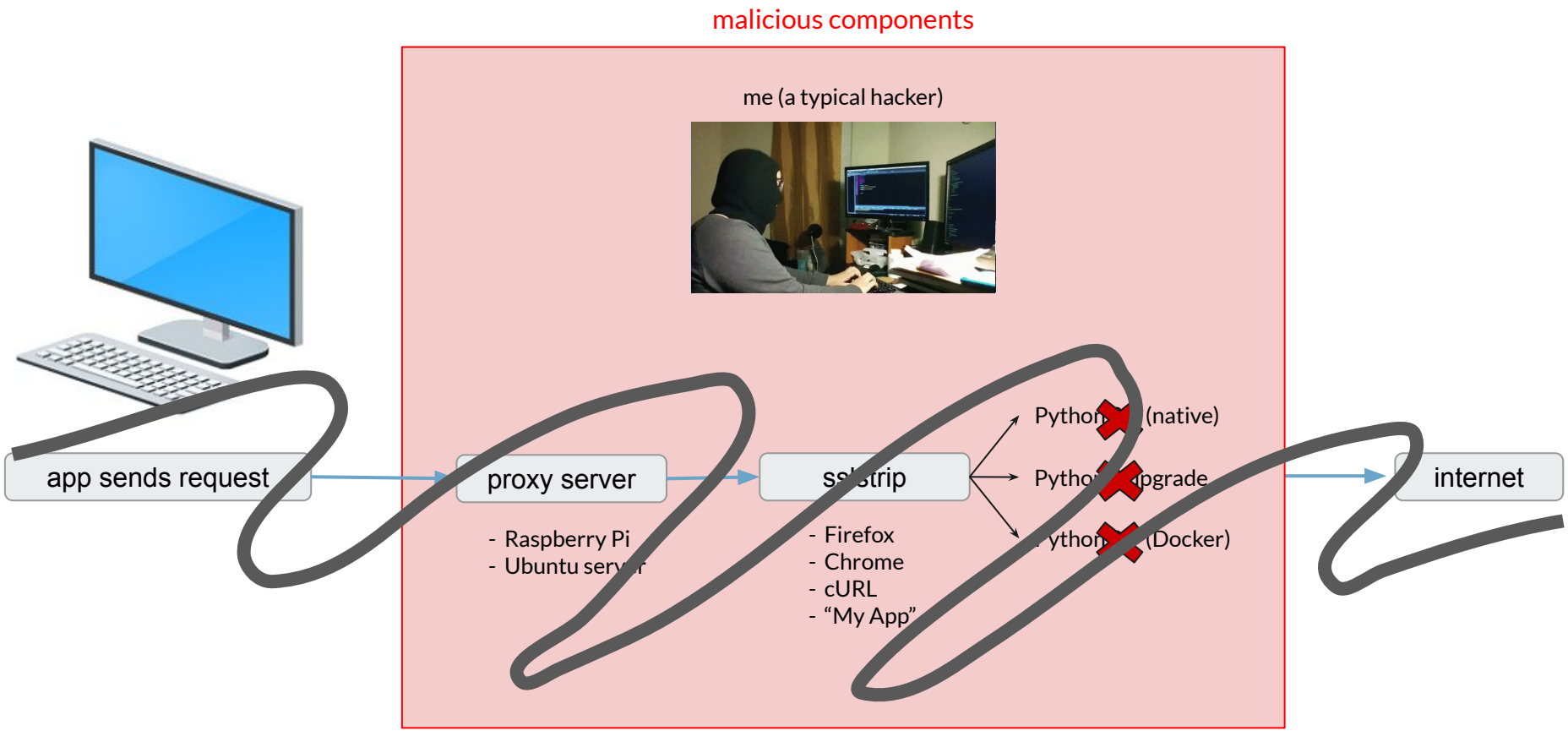
Step 1: Building an *OkHttp* test app for Android

Built my very first app ...

... to break it!



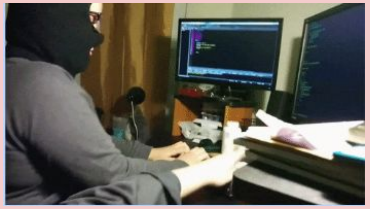
Step 2: Preparing an environment for sslstrip



Step 2: Preparing an environment for sslstrip (2nd take)

malicious components

me (a typical hacker)



establishing WLAN connection

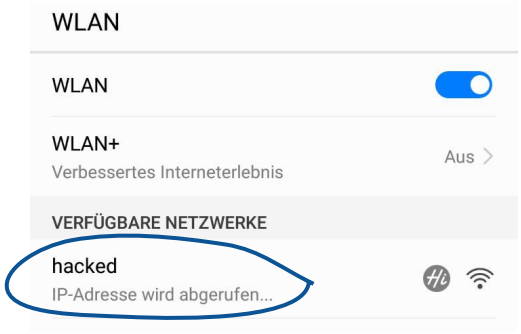
app sends request

Kali Linux with sslstrip

clie creds

ss tp2

internet



Lessons learned (SSL stripping)

SSL stripping is not trivial!

with Python 2.7: OpenSSL library is outdated

Handshakes do not work.

```
--- <exception caught here> ---
  File "/home/kali/.local/lib/python2.7/site-pac
    why = selectable.doRead()
  File "/home/kali/.local/lib/python2.7/site-pac
    return self._dataReceived(data)
  File "/home/kali/.
    rval = self.proto
  File "/home/kali/.
    why = self.rawDat
```

```
alex@alex-ThinkPad-T460p:~/dns2proxy$ python dns2proxy.py
Traceback (most recent call last):
  File "dns2proxy.py", line 21, in <module>
    import dns.message
  File "/usr/local/lib/python2.7/dist-packages/dns/message.py", line 186
    s.write(f';{name}\n')
                        ^
SyntaxError: invalid syntax
```

```
2020-12-18 09:42:34,432 Read from server:
wOF2◆
     ?◆◆0`◆T
◆L◆6
    ◆66$◆h ◆\◆
            2◆"◆8 ◆wE◆"◆◆◆npM$DDxUUÖo◆0#1◆g◆?_
◆>]+◆                                    p◆!◆/T◆◆◆X◆◆xLW
     ◆3Y,$◆◆◆◆◆6{◆36◆=◆◆J◆?◆◆H◆'+X9s◆◆◆v◆◆◆v◆Á◆◆z◆◆◆◆◆z◆◆^◆"◆h
◆q◆◆◆◆◆)◆ẍ◆r◆◆◆{        HE◆◆◆◆igKT◆Â0◆-◆◆%◆z´◆◆◆◆9◆|
A◆◆|X◆◆+%)◆*◆◆28%$◆nm◆\◆◆_e◆t5b̈◆n\◆G◆◆◆o◆2
                                        59◆   }C◆9◆◆K◆◆l*◆◆)j◆
`@◆F
                        D◆◆9*z,c◆◆◆jL=◆Ẋ◆◆◆◆&bS◆◆◆l,+◆◆9◆◆>*}◆◆@
15@◆◆◆◆◆Y◆◆◆xS◆◆;◆_◆_◆[◆%a◆&)◆raj◆◆
◆
◆◆S◆◆◆◆!-`{◆◆◆j#◆I
```

```
  File "/home/kali/sslstrip2/sslstrip/ServerConnection.py", line 141, i
    self.client.setHeader('Content-Length', len(data))
  File "/home/kali/.local/lib/python2.7/site-packages/twisted/web/http.
    self.responseHeaders.setRawHeaders(name, [value])
  File "/home/kali/.local/lib/python2.7/site-packages/twisted/web/http_
    for v in self._encodeValues(values)]
  File "/home/kali/.local/lib/python2.7/site-packages/twisted/web/http_
    return b' '.join(headerComponent.splitlines())
exceptions.AttributeError: 'int' object has no attribute 'splitlines'
```

SSLStrip+

This is a new version of [Moxie´s SSLstrip] (http://www.thoughtcrime.org/software/sslstrip/) with the new feature to avoid HTTP Strict Transport Security (HSTS) protection mechanism.

This version changes HTTPS to HTTP as the original one plus the hostname at html code to avoid HSTS. Check my slides at BlackHat ASIA 2014 [OFFENSIVE: EXPLOITING DNS SERVERS CHANGES] (http://www.slideshare.net/Fatuo__/offensive-exploiting-dns-servers-changes-blackhat-asia-2014) for more information.

For this to work you also need a DNS server that reverse the changes made by the proxy, you can find it at https://github.com/LeonardoNve/dns2proxy.

Demo video at: http://www.youtube.com/watch?v=uGBjxfizy48

BUT

Cause the new gag law which criminalized the publication of 'offensive' security tools/techniques I have to delete this repository. You can find good forks on MITMf framework (https://github.com/byt3bl33d3r/MITMf) or MANA rogue AP (https://github.com/sensepost/mana).

Lessons learned (Android)

HTTP is not supported by default

```
android:usesCleartextTraffic="true"
```

Missing *OkHttp* information:
What happens if TLS is not available?

OkHttp supports different TLS modes:

COMPATIBLE_TLS	A backwards-compatible fallback configuration that works on obsolete client platforms and can connect to obsolete servers. When possible, prefer to upgrade your client platform or server rather than using this configuration. `val COMPATIBLE_TLS: ConnectionSpec`
MODERN_TLS	A modern TLS configuration that works on most client platforms and can connect to most servers. This is OkHttp's default configuration. `val MODERN_TLS: ConnectionSpec`
RESTRICTED_TLS	A secure TLS connection that requires a recent client platform and a recent server. `val RESTRICTED_TLS: ConnectionSpec`

Blocked port 443 raises an error

Lessons learned (web browsers)

Modern web browsers have clever protections in place:

- Common website HSTS parametrizations are preloaded
- Chrome fakes a redirect for HSTS websites
- HTTPS to HTTP downgrades are only possible with the matching certificate
- HSTS leads to outages when expired certificates are used

Future work

Replication with other tools:

- Analysis of the *OkHttp* source code
- Evaluation of edge cases
- Try to use *(b)ettercap*
- Redirect to HTTP on network gateway

If vulnerabilities found, test them on other Android libraries and other projects:

- Github actions toolkit
- Microsoft's typed REST client (https://github.com/Microsoft/typed-rest-client)

Conclusion

On a personal note, I …

… learned a lot about web security and HSTS.

… got a deeper understanding of TLS.

… got to know a lot of handy tools.

… learned how to start developing apps!

Summary

What is SSL strip?

prevents secured connections

an “old” attack

mitigated by HSTS

Build an app

proper interface

OkHttp integration

Run a lot of scripts

evaluation of various tools

sslstrip / sslstrip2

easy-creds

bettercap

Failure analysis

(too) many dependencies

much technical knowledge required to fix errors