One Year of SSL Internet Measurement

ACSAC 2012

Olivier Levillain, Arnaud Ébalard, Benjamin Morin and Hervé Debar

ANSSI / Télécom SudParis

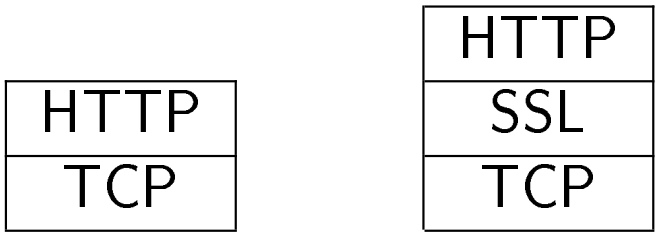
December 5th 2012

Outline

Outline

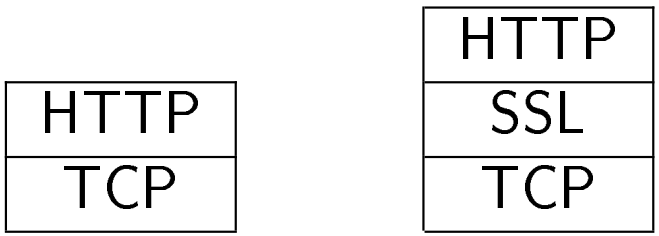
SSL/TLS : a key component of internet security

Originally, SSL/TLS is a transport layer between TCP and HTTP



SSL/TLS : a key component of internet security

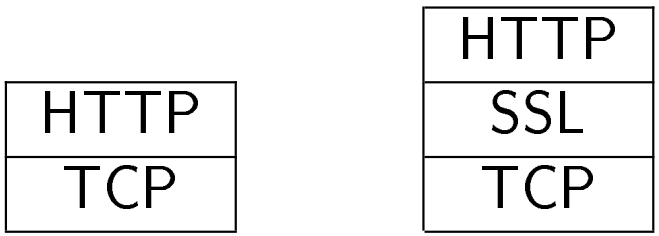
Originally, SSL/TLS is a transport layer between TCP and HTTP



- Security properties
  - Server authentication (or mutual authentification)
  - Data confidentiality
  - Data integrity

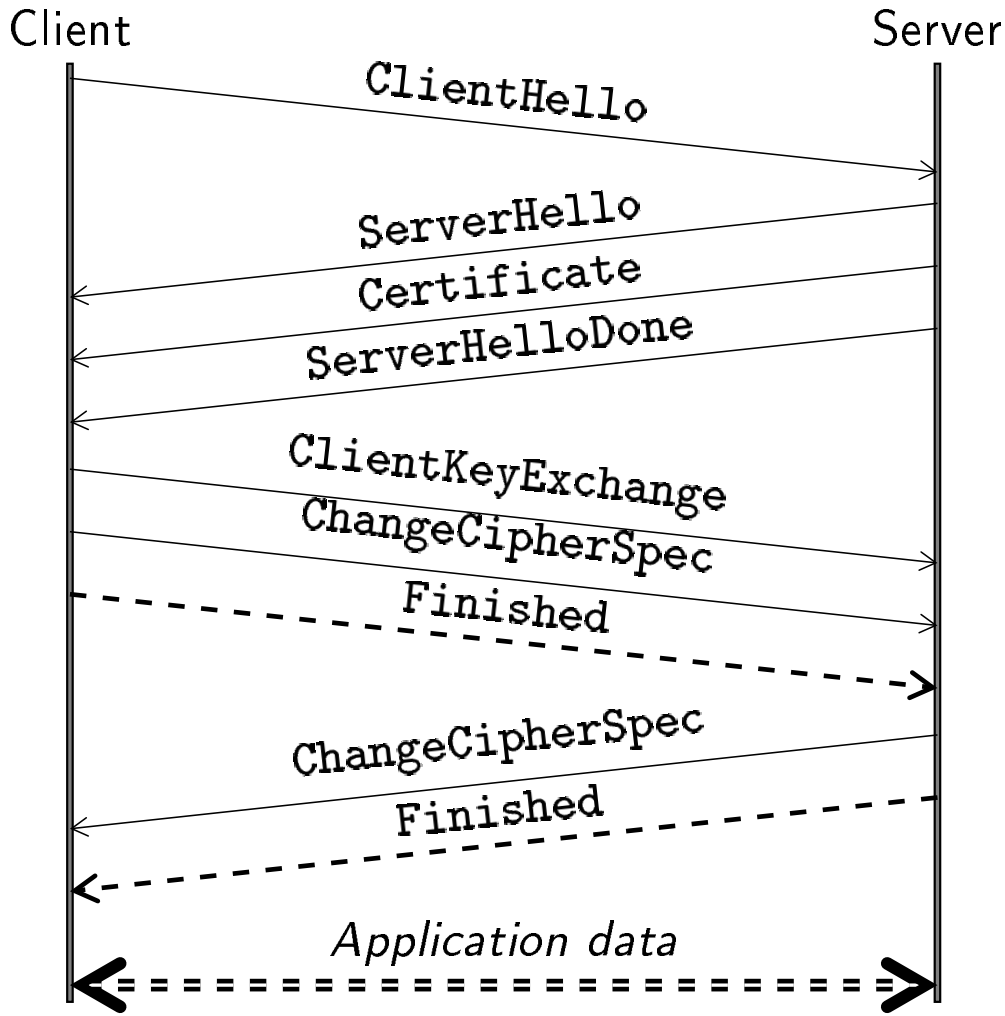
SSL/TLS : a key component of internet security

Originally, SSL/TLS is a transport layer between TCP and HTTP



- Security properties
  - Server authentication (or mutual authentification)
  - Data confidentiality
  - Data integrity
- Today, SSL/TLS is everywhere
  - SMTPS, IMAPS, LDAPS...
  - Virtual Private Networks
  - EAP-TLS

A typical TLS connection



Protocol history

- SSLv2, published by Netscape (1995)
- SSLv3, a major update to overcome SSLv2 structural flaws (1996)

- TLSv1.0
  - essentially SSLv3 with editorial changes (2001)
  - from this point, the protocol has been maintained by IETF
- TLSv1.1, which patches a cryptographic flaw (2006)
- TLSv1.2, which brings a little more flexibility (2008)

Known issues about SSL/TLS

- Protocol conceptual flaws
  - no handshake integrity (SSLv2)
  - insecure renegotiation (all versions before 2010)
  - information leak through compression (CRIME)

Known issues about SSL/TLS

- Protocol conceptual flaws
  - no handshake integrity (SSLv2)
  - insecure renegotiation (all versions before 2010)
  - information leak through compression (CRIME)

- Cryptographic weaknesses
  - short keys
  - PKCS#1 v1.5 implementation
  - Mac-Then-Encrypt implementation
  - Implicit IV in CBC mode (BEAST)

Known issues about SSL/TLS

- Protocol conceptual flaws
  - no handshake integrity (SSLv2)
  - insecure renegotiation (all versions before 2010)
  - information leak through compression (CRIME)

- Cryptographic weaknesses
  - short keys
  - PKCS#1 v1.5 implementation
  - Mac-Then-Encrypt implementation
  - Implicit IV in CBC mode (BEAST)

- Certificate problems
  - generation : lack of entropy
  - validation : null characters, wrongly used APIs
  - revocation : Comodo, Diginotar

Known issues about SSL/TLS

- Protocol conceptual flaws
  - no handshake integrity (SSLv2)
  - insecure renegotiation (all versions before 2010)
  - information leak through compression (CRIME)

- Cryptographic weaknesses
  - short keys
  - PKCS#1 v1.5 implementation
  - Mac-Then-Encrypt implementation
  - Implicit IV in CBC mode (BEAST)

- Certificate problems
  - generation : lack of entropy
  - validation : null characters, wrongly used APIs
  - revocation : Comodo, Diginotar

How to improve the quality of TLS connections ?

Outline

How to enumerate HTTPS hosts ?

- Enumerate every routable IPv4 address to find open HTTPS ports

How to enumerate HTTPS hosts ?

- Enumerate every routable IPv4 address to find open HTTPS ports

- Contact HTTPS hosts based on a list of DNS names

How to enumerate HTTPS hosts ?

- Enumerate every routable IPv4 address to find open HTTPS ports
- Contact HTTPS hosts based on a list of DNS names
- Collect real HTTPS traffic from consenting users

How to enumerate HTTPS hosts ?

- Enumerate every routable IPv4 address to find open HTTPS ports
  - EFF
- Contact HTTPS hosts based on a list of DNS names
  - Qualys SSL Pulse
  - NetCraft
- Collect real HTTPS traffic from consenting users
  - Holz et al., *The SSL Landscape — A Thorough Analysis of the X.509 PKI Using Active and Passive Measurements* (IMC '11)

How to enumerate HTTPS hosts ?

- Enumerate every routable IPv4 address to find open HTTPS ports
  - EFF
  - **our measures**
- Contact HTTPS hosts based on a list of DNS names
  - Qualys SSL Pulse
  - NetCraft
- Collect real HTTPS traffic from consenting users
  - Holz et al., *The SSL Landscape — A Thorough Analysis of the X.509 PKI Using Active and Passive Measurements* (IMC '11)

Our two-phase program

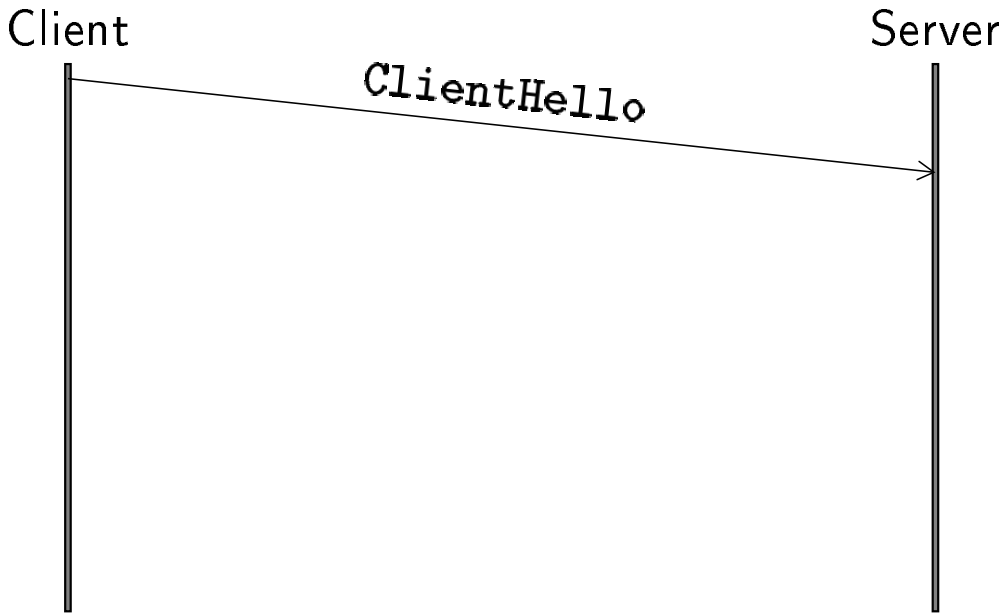
- Phase 1 : finding IPs with open TCP/443
  - 2 billion routable IPv4 addresses
  - randomisation of the set of addresses to contact
  - limited upstream rate to avoid links overloading
  - trade-off to get a time-coherent snapshot : 2 weeks, rate bounded at 100 kB/s

Our two-phase program

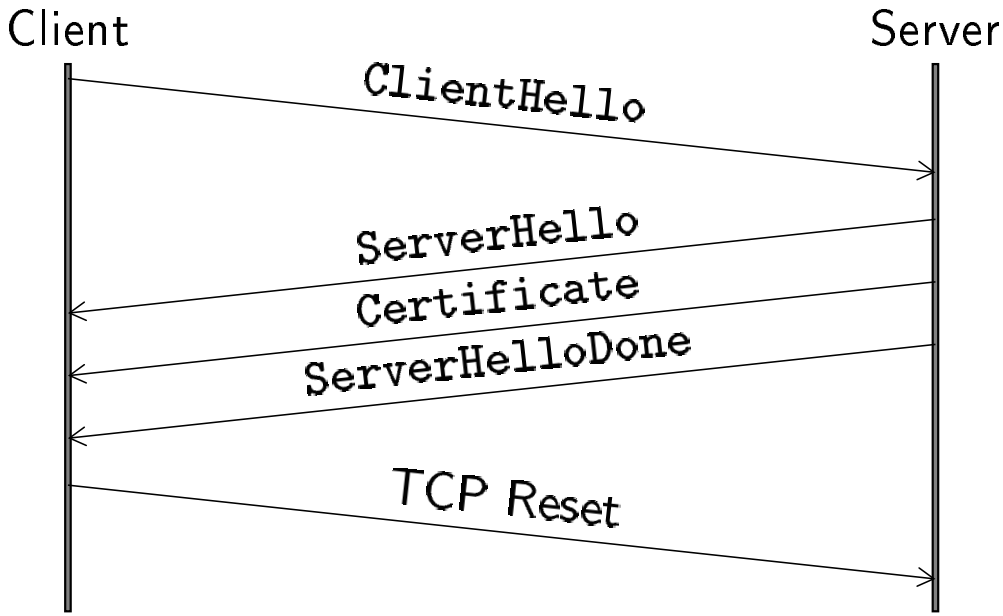
- Phase 1 : finding IPs with open TCP/443
  - 2 billion routable IPv4 addresses
  - randomisation of the set of addresses to contact
  - limited upstream rate to avoid links overloading
  - trade-off to get a time-coherent snapshot : 2 weeks, rate bounded at 100 kB/s
- Phase 2 : TLS session attempt
  - about 1 % of hosts have TCP/443 open
  - description of the message exchanged
    - we send a `ClientHello` (the stimulus)
    - we gather the answer, at most until the `ServerHelloDone`
    - we send a TCP Reset

The different types of answers

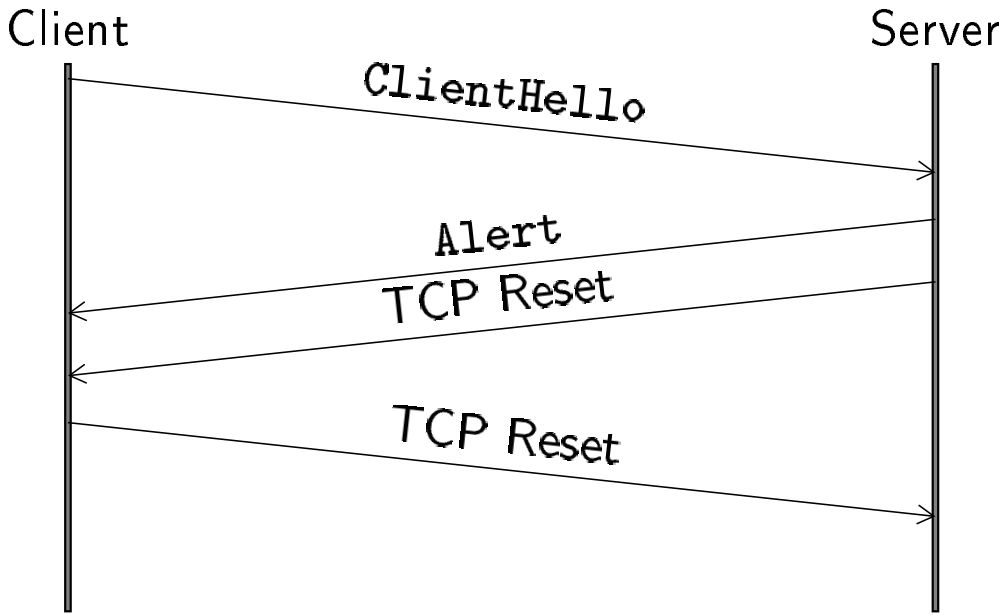
The different types of answers



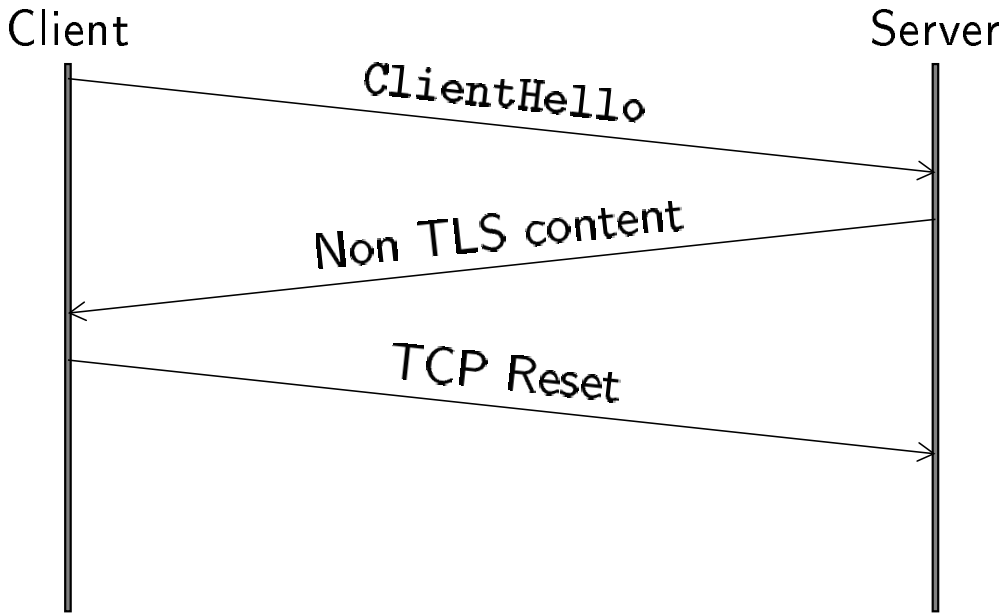
The different types of answers



The different types of answers



The different types of answers



Description of the 10 campaigns

Id	Date	SSLv2	Max version	Ciphersuites	Extensions
NoExt1	2010/07	no	TLSv1.0	Firefox	None
EFF-1	*2010/08*	*yes*	*TLSv1.0*	*SSLv2 + TLSv1*	*None*
EFF-2	*2010/12*	*yes*	*TLSv1.0*	*SSLv2 + TLSv1*	*None*
NoExt2	2011/07	no	TLSv1.0	Firefox	None
DHE	2011/07	no	TLSv1.0	DHE Suites	None
FF	2011/07	no	TLSv1.0	Firefox	EC, Reneg, Ticket
EC	2011/07	no	TLSv1.0	EC Suites	EC
SSL2	2011/07	yes	SSLv2	SSLv2	None
SSL2+	2011/07	yes	TLSv1.0	SSLv2 + TLSv1	Reneg
TLS12	2011/07	no	TLSv1.2	Mostly TLSv1.2	EC, Reneg, Ticket

Those last 7 stimuli were sent in parallel to study in detail the server behaviour.

Global statistics

Id	IPs with TCP/443	Non-TLS answers	TLS answers
NoExt1	21,342,205	54 %	46 %
EFF-1	*15,579,266*	*27 %*	*73 %*
EFF-2	*7,777,511*	*1 %*	*99 %*
NoExt2	26,218,653	57 %	43 %
DHE	26,218,653	66 %	34 %
FF	26,218,653	57 %	43 %
EC	26,218,653	64 %	36 %
SSL2	26,218,653	81 %	19 %
SSL2+	26,218,653	57 %	43 %
TLS12	26,218,653	64 %	36 %

Outline

Subsets

For each campaign, we consider 3 subsets :

- TLS hosts
- Trusted hosts (using Firefox certificate store)
- EV hosts

Studied critera

- TLS parameters
  - protocol version chosen by the server
  - ciphersuite selected by the server
  - secure renegotiation support

Studied critera

- TLS parameters
  - protocol version chosen by the server
  - ciphersuite selected by the server
  - secure renegotiation support

- Quality of the certification chain
  - `Certificate` message analysis
  - key sizes
  - validity periods

Studied critera

- TLS parameters
  - protocol version chosen by the server
  - ciphersuite selected by the server
  - secure renegotiation support
- Quality of the certification chain
  - `Certificate` message analysis
  - key sizes
  - validity periods
- Server behaviour
  - version intolerance
  - ciphersuite intolerance

Outline

Protocol version

For a typical campaign (NoExt1, EFF1, EFF2, NoExt2, FF), the version chosen are stable in time :

TLS		**Trusted**		**EV**	
TLS1	96 %	TLS1	99 %	TLS1	99 %
SSL3	4 %	SSL3	1 %	SSL3	1 %

Secure Renegotiation extension (RFC 5746)

- Only 3 stimuli proposed the extension
- All in 2011, so we can not observe a trend
- In the three cases, the proportion of servers accepting the extension is the same

TLS hosts	53 %
Trusted	65 %
EV	80 %

The Certificate message

The RFC indicates that

- all the certificates of the chain should be present
- in the order of the chain
- the root may be ommited

In practice, we saw four types of chains

- RFC-compliant
- Self-contained
- Transvalid
- Incomplete

Evolution of the types of the chains

	2010-07	2010-08	2010-12	2011-07
TLS	**R : 60 %** S : 9 % T : 4 % I : 27 %	**R : 61 %** S : 8 % T : 3 % I : 28 %	**R : 59 %** S : 10 % T : 6 % I : 25 %	**R : 54 %** S : 10 % T : 6 % I : 30 %
Trusted	**R : 69 %** S : 21 % T : 10 %	**R : 71 %** S : 19 % T : 10 %	**R : 67 %** S : 21 % T : 12 %	**R : 62 %** S : 24 % T : 14 %
EV	**R : 11 %** S : 78 % T : 11 %	**R : 13 %** S : 76 % T : 11 %	**R : 16 %** S : 74 % T : 10 %	**R : 12 %** S : 83 % T : 5 %

Some figures about certificates

- RSA is by far the main signature algorithm used in certs
- Typical RSA key size was 1024 bits in 2010 but 2048 bits is now mandatory for EV certificates

Some figures about certificates

- RSA is by far the main signature algorithm used in certs
- Typical RSA key size was 1024 bits in 2010 but 2048 bits is now mandatory for EV certificates
- but we saw 16384 bits or 384 bit keys in the measures
- 512 bit trusted certs still found in July 2011

Some figures about certificates

- RSA is by far the main signature algorithm used in certs
- Typical RSA key size was 1024 bits in 2010 but 2048 bits is now mandatory for EV certificates
- but we saw 16384 bits or 384 bit keys in the measures
- 512 bit trusted certs still found in July 2011

- Typical `Certificate` message contains at most 3 certs

Some figures about certificates

- RSA is by far the main signature algorithm used in certs
- Typical RSA key size was 1024 bits in 2010 but 2048 bits is now mandatory for EV certificates
- but we saw 16384 bits or 384 bit keys in the measures
- 512 bit trusted certs still found in July 2011

- Typical `Certificate` message contains at most 3 certs
- but one trusted host sent 150 certificates in 2010

Some figures about certificates

- RSA is by far the main signature algorithm used in certs
- Typical RSA key size was 1024 bits in 2010 but 2048 bits is now mandatory for EV certificates
- but we saw 16384 bits or 384 bit keys in the measures
- 512 bit trusted certs still found in July 2011

- Typical `Certificate` message contains at most 3 certs
- but one trusted host sent 150 certificates in 2010

- Typical validity period is one or two years

Some figures about certificates

- RSA is by far the main signature algorithm used in certs
- Typical RSA key size was 1024 bits in 2010 but 2048 bits is now mandatory for EV certificates
- but we saw 16384 bits or 384 bit keys in the measures
- 512 bit trusted certs still found in July 2011

- Typical `Certificate` message contains at most 3 certs
- but one trusted host sent 150 certificates in 2010

- Typical validity period is one or two years
- but some certs are valid until 9999
- and others never were (`notBefore` $>$ `notAfter`)

Server behaviour

- We now consider the 7 stimuli sent in July 2011 essentially at the same time
- Based on the certificates returned, we are confident the hosts contacted were stable accross the 7 answers
- Redefine our subsets :
  - TLS hosts are hosts that spoke TLS at least once
  - Trusted hosts are hosts that returned a trusted chain at least once
  - Same thing for EV hosts

The DHE stimulus

- DHE stands for Diffie-Hellman Ephemeral
- DHE provides Perfect Forward Secrecy
- The DHE stimulus only proposed DHE ciphersuites

	TLS	**Trusted**	**EV**
Compatible Handshake	39 %	42 %	13 %
Alert	38 %	28 %	71 %
Intolerant servers	**23 %**	**30 %**	**16 %**
Non-TLS answer	22 %	30 %	16 %
Incompatible Handshake	1 %	0 %	0 %

The TLS12 stimulus

- The TLS12 stimulus proposed versions TLSv1.0 to TLSv1.2
- Servers can answer with TLSv1.0 if they don't know TLSv1.2 (and they should, because it is part of the negotiation)

	TLS	**Trusted**	**EV**
Compatible Handshake	76 %	74 %	86 %
Alert	7 %	5 %	2 %
Intolerant servers	**17 %**	**21 %**	**12 %**
Non-TLS answer	16 %	21 %	12 %
Incompatible Handshake	1 %	0 %	0 %

Outline

Conclusion

- Study of the IPv4 HTTPS landscape from July 2010 to July 2011
- Simultaneous stimuli in July 2011, allowing to observe the server behaviour (more complex as it seems, Google's False Start)
- Different subsets and different times to show some trends
- Studied criteria were not only about certificates
- Lots of surprising answers

Conclusion

- Study of the IPv4 HTTPS landscape from July 2010 to July 2011
- Simultaneous stimuli in July 2011, allowing to observe the server behaviour (more complex as it seems, Google's False Start)
- Different subsets and different times to show some trends
- Studied criteria were not only about certificates
- Lots of surprising answers

- EV is a certificate label and has a clear impact on RSA key sizes and certificate validity periods
- However, on all other criteria, EV hosts behave poorly (they are even worse than the global TLS statistics in some cases)
- Need for a label attesting the global quality of TLS connections

Future work

- More criteria to study
  - more TLS parameters (DH groups, revocation mechanisms, other extensions)
  - take HTTP parameters into account (mixed content)

Future work

- More criteria to study
  - more TLS parameters (DH groups, revocation mechanisms, other extensions)
  - take HTTP parameters into account (mixed content)
- New campaigns
  - use real navigation data
  - contact the HTTPS hosts identified and inspect them thoroughly

Questions ?

This work has been partially sponsored by the EC 7th Framework Programme as part of the ICT Vis-Sense project (grant no. 257497)

Thank you for your attention