

Alternatives to PKI-based SSL on the web

Dr. István Zsolt Berta

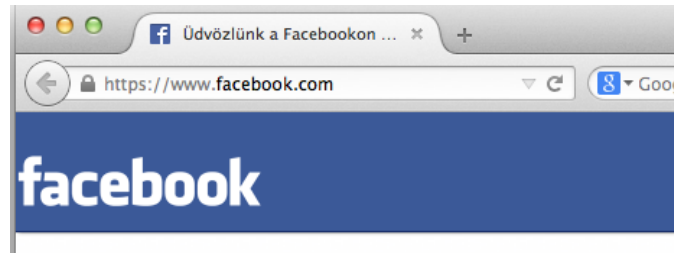
www.berta.hu

istvan@berta.hu

opinions expressed here
are strictly those of my own

Alternatives to PKI-based SSL on the web

- A https connection means you are communicating with the website in the URL, the connection is encrypted and no one else can tamper with it



- Security is based on an SSL certificate issued by a trusted Certificate Authority
- In this talk, we shall examine and evaluate any alternative approaches that exist

Recent problems with SSL

- Issues with the security of Certificate Authorities
 - Comodo, Diginotar, KPN, Trustwave, ... (see more info [here](#))
- News on international espionage
 - attacks against CAs
 - [compelled certificate attack](#) (i.e. a government orders a CA to issue a false certificate)
- Weaknesses in the protocol
 - [renegotiation](#), BEAST, CRIME, etc.
- Weaknesses in SSL implementations
 - [gotofail](#), [heartbleed](#), [CSS injection](#), etc.
- Weak SSL keys [in large numbers](#) (0.2% of all keys on the web)

Initiatives for improving CA security

- [CA/Browser Forum](#)
 - industry-led attempts to make order and improve security
 - [Baseline Requirements](#)
 - [Network Security Req](#)s
 - all are very basic requirements
 - how are they enforced?
- New EU regulation replacing the e-Signature Directive
 - more focus on security
 - focus on incident reporting
 - will apply to SSL certificates too (current Directive is for e-signature only)

Regardless of these initiatives...

- Browsers trust all (100+) CAs globally; if one CA is breached, the attacker can impersonate any website
- CAs operate in different countries and jurisdictions, these trust each-other... but to a certain level only
 - Are we trying to establish a trust relationship electronically that does not exist in the real world?
- Commercial CAs
 - will always be driving down costs to stay competitive
 - select the auditor they prefer
- Governmental CAs
 - often do not have a proper, independent audit, but provide an audit-equivalency statement only



Approach: Let's have fewer CAs

- Why are we trusting 100+ CAs, where some are very small and are from distant countries you have never heard of? Most certs are issued by a few global CAs; why trust small ones?
 - Smaller countries would need to rely on security from someone else – will they accept this?
 - Recent news on attacks include: [Comodo](#), [Verisign](#), [Globalsign](#)...
Hey, these are the big ones!!!
- Still, if you know that you need a few CAs in a certain application only, there can be point in distrusting all others



Approach: Let's restrict the authority of CAs

- Why are all CAs trusted globally? Why are not they restricted to e.g. a country/region, etc?
- Yes, but we now have global CAs – what to do with them?
- Who would be limiting the market and how?
- X.509 has a plethora of tools for this (Name Constraints, Policy Constraints, etc)
 - We are still having problems around Basic Constraints (differentiating CA and end-entity certs) in browsers
 - X.509 path building is VERY complex, hard to do well
- CA/Browser Forum documents allow CAs to constraint themselves voluntarily – browsers do not support it yet
- Still, this could be a way forward...



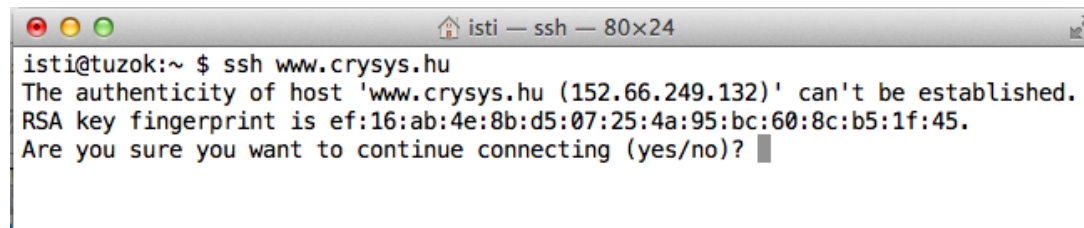
Self-signed certificates

- The connection is encrypted and integrity checks are applied but you do not know who you are connected to
- They provide no protection against man-in-the-middle attacks
- Considered as heresy
- But: Certificates are used when verifying if the given public key belongs to the given entity (web server) only; what if I do this check myself?
 - Example: I receive the cert on a secure channel
 - Example 2: Check cert fingerprint with the counterpart
 - Some people actually [try to do this](#)...
 - Come on, this approach does not scale!!



Approach: Trust on First Use (TOFU)

- First time you receive the key → trust it; but be suspicious when it changes
- SSH uses the same concept – who checks the fingerprint? (yes, but SSH is not used towards arbitrary servers globally)

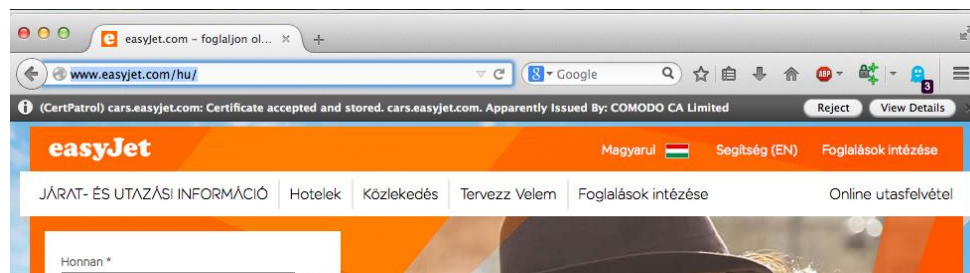


```
isti@tuzok:~ $ ssh www.crysys.hu
The authenticity of host 'www.crysys.hu (152.66.249.132)' can't be established.
RSA key fingerprint is ef:16:ab:4e:8b:d5:07:25:4a:95:bc:60:8c:b5:1f:45.
Are you sure you want to continue connecting (yes/no)?
```

- No protection against man-in-the-middle attacks on first use; but if there is a MITM attack on first use, the attacker must remain in the connection (forever) or risk being detected
- Phil Zimmermann's [ZFone](#) uses a similar approach: [RFC 6189](#)

Tool: Certificate Patrol

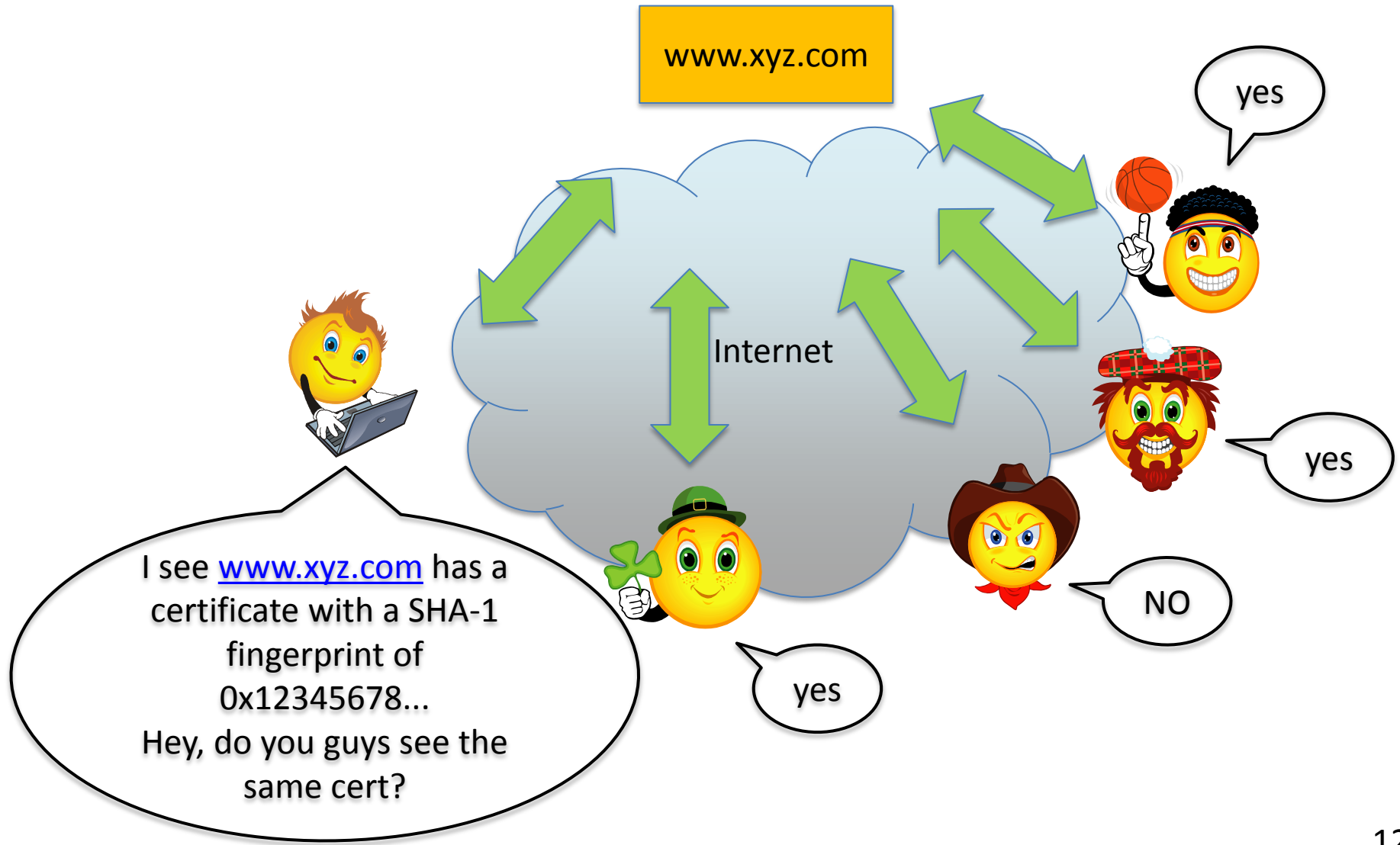
- A [Firefox Addon](#) implementing certificate pinning
- Takes note of certificates of sites you visit
- For known sites, checks if the certificate is known
- Displays a warning message when a site's certificate changes
- Provides a different treatment for low-threat harmless-looking updates (e.g. same key? same CA?)



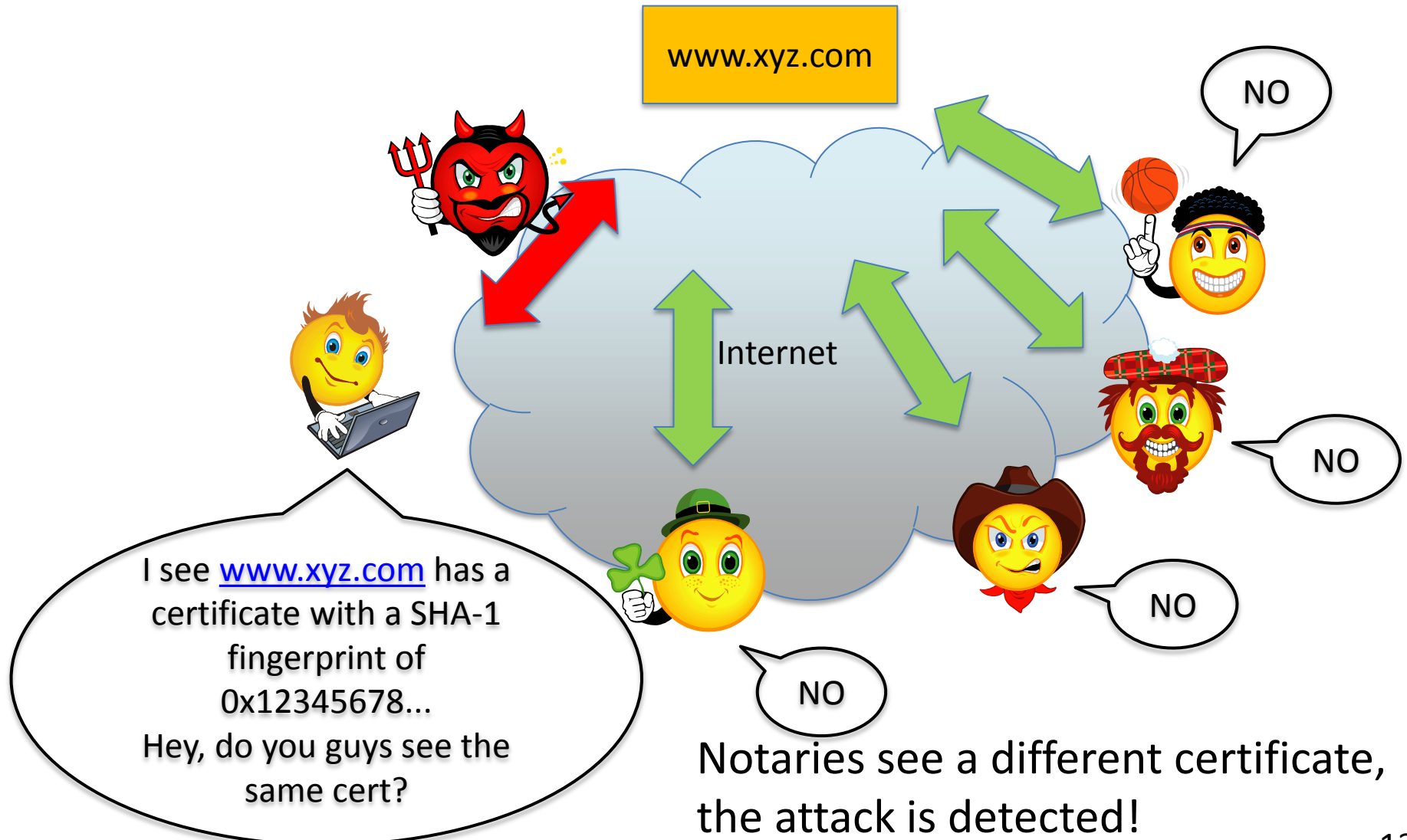
Tool: Perspectives

- Relies on multiple network notaries who continuously monitor public keys used by webserver
- When the client connects to a new website, she contacts some randomly selected notaries and asks what public keys they see
- The website is looked at from different *perspectives*, i.e. by the client and by the notaries
- Uses PGP for protecting communication with notaries
- Also incorporates the TOFU approach, contacts notaries when a key/cert is updated only
- Client is available as [Firefox Addon](#)
- Research paper: [Wendlandt&Andersen&Perrig, 2011](#) (CMU)

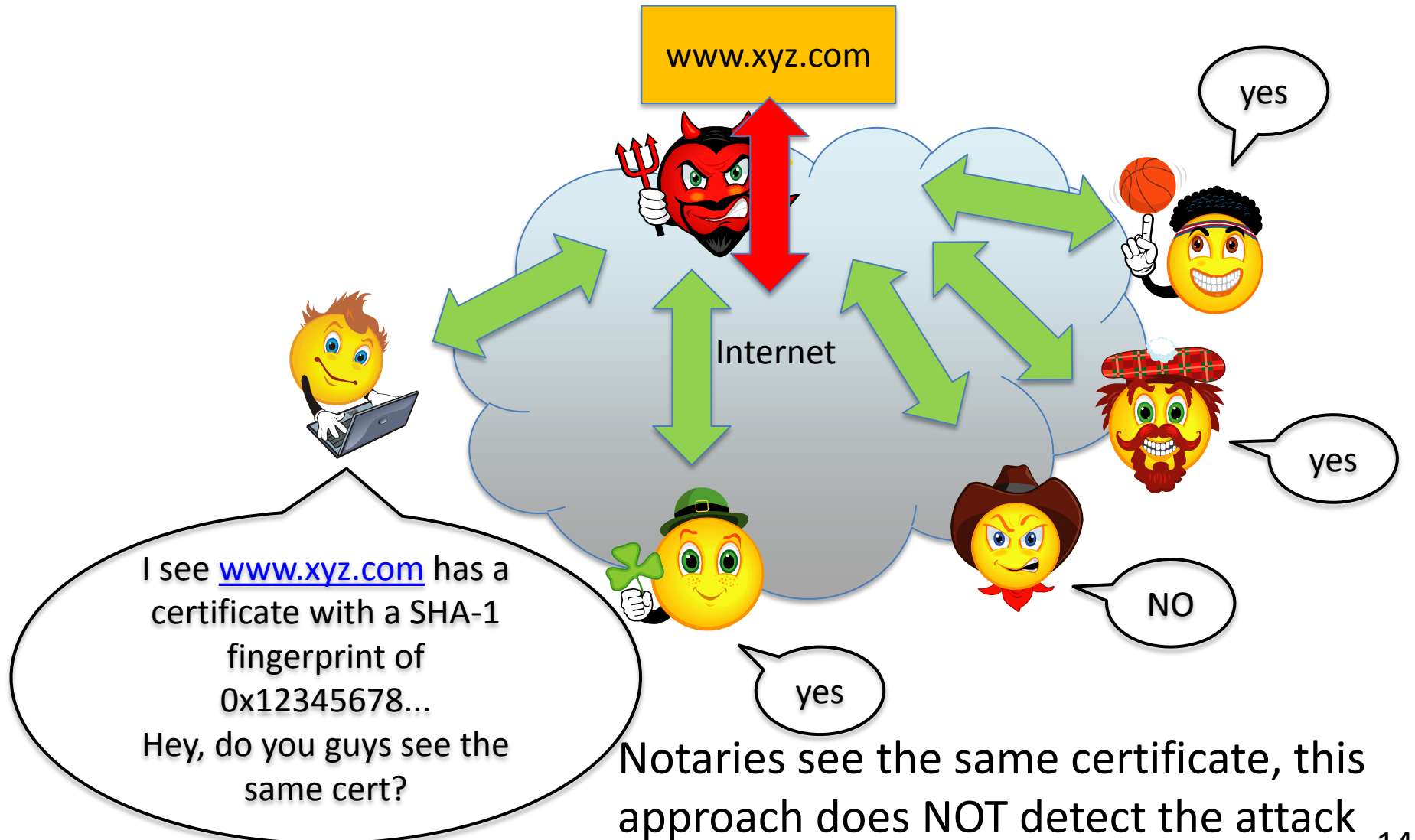
How Perspectives works



Perspectives – Client ISP is Evil



Perspectives – Server ISP is Evil



Notes on TOFU and networked verification

- The Diginotar incident was [detected](#) by a user who saw a different and unknown CA as the issuer of GMail.com
- These approaches struggle if the site's certificate changes quickly legitimately
 - for instance, if a site is supported by multiple servers (for balancing the load) that have different certificates (because each server has a different key pair)

Tool: Convergence

- An extension of Perspectives, by Moxie Marlinspike
- Crowdsourcing the networked verification, i.e. all clients also act as notaries
- More control over votes from notaries (consensus, majority vote, etc.)
- Uses onion routing for anonymous connections to notaries
- <http://convergence.io/>, [Firefox Addon](#)

Summary of concepts presented

- TOFU & Identity change detection (certificate pinning)
 - provides forward secrecy
 - example: Certificate Patrol
- Networked verification of identity
 - works if the man-in-the-middle attack is targeted at a client, and not at the whole web
 - example: Perspectives, Convergence
- Encrypting / Authenticating the connection based on the key obtained the above way, via regular SSL

Conclusions

- There is no major problem with SSL and web-based PKI
- Of course, you should not trust it blindly, it has limitations
- SSL provides sufficient protection against most attackers, but does not help against those few who can tamper with CAs
- Identity change detection and network verification of identity approach the problem differently, they can be viable
- I do not think any of the presented tools/approaches are significantly better than PKI-based SSL, they are cheaper but (probably) have a lower level of security
- Security geeks can combine these currently immature tools with PKI-based SSL to gain more security



Bonus: When using SSL in an automated system

- Use a proper tool for performing the PKI-based verification of the certificate of your counterpart, do not write your own
- Remove/Distrust all CAs you do not need
- Apart from the PKI-based verification there might be point in checking the following for your counterpart's certificate
 - Subject DN / Issuer DN, and/or
 - fingerprint (this needs to be updated at each certificate change, so e.g. every two years)

Thank you very much!

Dr. István Zsolt Berta

www.berta.hu

istvan@berta.hu

Alternatives to PKI-based SSL on the web

Dr. István Zsolt Berta

www.berta.hu

istvan@berta.hu

opinions expressed here
are strictly those of my own