Operation Black Tulip: Certificate authorities lose authority

DigiNotar, a digital certificate authority (CA), recently suffered a cyber-attack which led to its bankruptcy. In the attack false certificates were created for hundreds of websites, including Google and Skype. Once the incident was made public, the Dutch government and browser vendors took steps to limit the impact of the attack. But Fox-IT suggests in their investigation report that the cyber-attack had already started in mid-June and that for almost two months false certificates were used to eavesdrop on email and web browsing in Iran. We see three major issues:

1. **No immediate incident reporting:** DigiNotar did not immediately report the cyber-attack to customers or government authorities, which put the security and privacy of millions of citizens at risk. Immediate reporting of the incident and a swift response would have limited the impact considerably.
2. **Fundamental weaknesses in the design of HTTPS:** In the current setup, browsers and operating systems (e.g. Microsoft's certificate store) place trust by default in a large number of CAs (hundreds) by default, so a failure with one of them creates a risk for all users and all websites. The security of HTTPS equates to the security of the weakest CA. HTTPS should be modernized, to be more resilient against attacks and more user-friendly.
3. **Failure to implement basic security measures:** The Fox-IT report shows that basic security measures were not taken[1]. It is imperative that service providers, like CAs, which play such a critical role in today's digital society, adhere to best practices. The attack highlights the importance of enforcing basic security best practices.

The Diginotar attack was an attack on the foundations of secure electronic communications (email, web browsing, web services). The above-mentioned issues should be addressed by industry and governments, to guarantee the security of service in the digital society.

Technical background

SSL (Secure Sockets Layer) certificates are used in the https protocol to secure digital communications, such as web browsing, email and machine-to-machine communications (web services), and to create electronic signatures. DNSSEC (Domain Name System Security) also relies on SSL certificates. False certificates can be used for example to intercept private emails, execute fraudulent banking transactions, or create false digital signatures.

The Fox-IT investigation report suggests that the false certificates were used to perform a large-scale MITM (man-in-the-middle) attack on users in Iran[2]. For a

[1] Diginotar was audited yearly by an independent auditor against the ETSI standard (TS101456) for certificate authorities.

[2] According to FoxIT, Diginotar's logs showed 300.000 OCSP (Online Certificate Status Protocol) requests from IP addresses in Iran. Usually browsers make OCSP requests to check whether a certificate has been revoked or not. This suggests that around 300.000 Iranian users were victims of MITM attacks. But the OCSP requests are only an indication, because not all browsers or clients make OSCP requests, and because OSCP requests could have been blocked or faked by attackers.

MITM attack, besides false SSL certificates, the attacker needs to be able to intercept and modify IP traffic. In general this can be done by using a rogue hotspot, by poisoning the DNS cache or ARP cache, by using malware on the victim's machine, or by accessing the traffic at ISPs directly.

Diginotar did not have a record of all the rogue certificates that were created by the attacker, so the only remedy was to remove the root certificate of Diginotar from all the browsers. This was a major issue for many websites in the Netherlands. But if the same had happened to a large CA, the problem would have been even more serious. To give an example, in March 2011 the CA, Comodo, which has roughly a quarter of the global SSL market, was the victim of a similar attack. Revoking trust in a CA this large would have had serious consequences on a global scale: It can even be argued that CAs of this size are too large to fail.

Lack of incident reporting: The inadequacies in reporting the incident to customers and government authorities appears to have had serious consequences: for 2 months, private communications could be intercepted. It has been argued that the delay in incident reporting put lives at risk[3]. This incident is a clear illustration of the importance of incident reporting schemes.

Poor security practices: Fox-IT reports that there was no antivirus protection in place, weak administrator passwords and insufficient logging. This is despite the fact that Diginotar was audited periodically to ETSI standards required for Extended Validation certificates and Qualified Electronic Signatures[4].

Weaknesses in the implementation and design of HTTPS: Beyond the single case of Diginotar, this attack shows up weaknesses in the design and implementation of HTTPS.

- **Weakest link:** The web browser 'trusts' HTTPS connections if the certificate is issued by any of the (about 600) CAs in the trusted list of the web browser. If any one of these CAs is vulnerable to attack then this poses a risk to all users and all websites. This is because, for a given site, browsers will accept a valid certificate from any of these CAs. Therefore an attacker who has compromised any CA in the browser's list can create a valid certificate for *any* web site which will be accepted by the browser. In this sense, each CA is a single point of failure. Larger CAs work with hundreds of resellers, increasing the attack surface even more.
- **Weak revocation check:** CAs have OCSP (Online Certificate Status Protocol) responders which can be queried by a user's browser or a web service client, to check whether a certificate has been revoked or not. OSCP can limit the impact of false or rogue certificates, but often browsers and web services clients do not use OCSP or do not warn the user clearly when an OCSP check has failed.
- **Usability issues**: HTTPS warnings by users are often ignored[5] and the *absence* of a warning is often the only indication that HTTPS is not being used. Decisions about HTTPS are left to users who get frequent false alarms and are unable to understand the warnings. Also, users with high security requirements (for example, politicians, political activists, etc.) cannot switch to a more

[3] Van Dam, a Dutch MP said the actions of Diginotar had put lives at risk, and Mikko Hypponen, from F-Secure, argued in the press that activists have died as a consequence of the delayed reporting about the attack.

[4] The Diginotar website until recently showed an audit report stating that "the management system for issuance of certificates of DigiNotar complies with ETSI TS 101 456 (v. 1.4.3) - normalized certificate polices NCP+, EV specified in ETSI TS 102 042 (v. 2.1.2)."

[5] This is demonstrated by the case of New Zealand's BankDirect which accidentally allowed a certificate to expire. Server logs show that all but one of 300 users dismissed the HTTPS security warning.

secure CA. Because users have no choice in the matter, websites have little incentive to choose more secure CAs.

Addressing the issues

We suggest the following steps to address the above-mentioned issues.

- **Incident reporting:** Companies, like CAs, who deliver important digital society services, should pro-actively detect and investigate incidents and quickly inform the relevant parties (users involved, corporate customers involved, government authorities) about significant security incidents. Industry and government should agree on incident reporting schemes for crucial service providers in the digital society. The Dutch government plans to adopt an obligation for security incident reporting. EU Commissioner Kroes indicated in the aftermath of this attack that the reporting of security incidents may become obligatory for CAs. Current European law requires only telecommunications providers to report security and privacy breaches.
- **Security best practices:** Service providers who play a crucial role in the digital society should implement security best practices. Industry and government should agree on ways to ensure that service providers comply with security best practices. Existing certification and audit schemes may need to be revised considering that Diginotar was audited yearly against the ETSI standards for CAs.
- **Improved implementation of HTTPS:** Currently the security of HTTPS depends on the security of the weakest CA included in the browser list. Browser vendors should ensure that these CAs can be trusted and if needed remove insecure CAs from their trusted lists. OSCP should be implemented better – i.e. browsers should take a pessimistic approach and give the user clear warnings when a CA does not reply to an OSCP request, or when a CA does not confirm a certificate is valid (for example, if a CA answers 'revoked' or 'unknown').
  Additional protection could be offered by techniques like CA pinning, whereby the website specifies, on the first visit, which CA it uses. Another proposal is DANE (DNS-based Authentication of Named Entities) which associates the valid certificate with the authoritative DNS record (although this relies on the implementation of DNSSEC). We encourage browser vendors and other stakeholders to discuss and agree on such implementation fixes to shore up the security of HTTPS.
- **Better design of HTTPS:** The attack on Diginotar (like the hack on Comodo some months ago) also raises doubts about the fundamental design of HTTPS. It is good to keep in mind that HTTPS was developed when there were few websites, only a handful of CAs, and hardly any secure sites. Today there are millions of secure websites and CAs face the difficult task of verifying the identity of millions of web domains. There have been several proposals for alternative implementations of HTTPS. Convergence, for example, offers an alternative for CA's as the sole provider of trust in SSL certificates (adding support for network perspectives, and SSL Observatory, for example) which could be more robust against cyber-attacks. We look forward to seeing the IT industry and browser vendors agree on a more secure and user-friendly design of HTTPS.