Academisch Schrijven Opdracht 3

Thijs klein Baltink, s4359763

1 Article

https://drownattack.com/drown-attack-paper.pdf

a The target audience of this article consists of Computer scientists, cryptographers, or other people with an interest in, and background knowledge of Transport Layer Security (TLS).

b The goal of the authors seems to be to make it clear that SSLv2 is weak and harmful to TLS as a whole, they do this by presenting 2 attacks using an SSLv2 server.

c The abstract is reasonably well written; it clearly states the attacks that will form the body of the article and the conclusion that is drawn from this, however the abstract consists of too many statistics. These might help to make the article seem like a must read, but it does detract from the abstract; it left no room for any introduction within the abstract and also left only minor detail of their conclusion. The abstract is effective at making the article seem important, but only moderately effective at detailing the content of the article.
This introduction also touches on the history of TLS, then goes to further detail the history concerning the SSLv2 protocol. They continue by detailing Bleichenbacher's padding oracle attack concluding the first segment of the introduction. In the second segment the article further details the attacks that the authors used on SSLv2, followed up by statistics of what exactly this would affect, to be concluded with a condensed version of their conclusion. This introduction does a good job at introducing the context and content of the article, however it does seem to foreshadow to an unnecessary degree to the body and conclusion of the article. The article seems rather effective, however the somewhat excessive description of the body and conclusion of the article are detrimental to the brevity of the introduction.

2 abstract

We present DROWN, a novel cross-protocol attack on TLS that uses a server supporting SSLv2 as an oracle to decrypt modern TLS connections.
We introduce two versions of the attack. The more general form exploits multiple unnoticed protocol flaws in SSLv2 to develop a new and stronger variant of the Bleichenbacher RSA padding-oracle attack. To decrypt a 2048-bit RSA TLS ciphertext, an attacker must observe 1,000 TLS handshakes, initiate 40,000 SSLv2 connections, and perform 2 50 offline work.
The victim client never initiates SSLv2 connections.
We implemented the attack and can decrypt a TLS 1.2 handshake using 2048-bit RSA in under 8 hours, at a cost of $440 on Amazon EC2. Using Internet-wide scans, we find that 33% of all HTTPS servers and 22% of those with browser-trusted certificates are vulnerable to this protocol-level attack due to widespread key and certificate reuse. For an even cheaper attack, we apply our new techniques together with a newly discovered vulnerability in OpenSSL that was present in releases from 1998 to early 2015. Given an unpatched SSLv2 server to use as an oracle, we can decrypt a TLS ciphertext in one minute on a single CPU—fast enough to enable man-in-the-middle attacks against modern browsers. We find that 26% of HTTPS servers are vulnerable to this attack.

We further observe that the QUIC protocol is vulnerable to a variant of our attack that allows an attacker to impersonate a server indefinitely after performing as few as 217 SSLv2 connections and 258 offline work. We conclude that SSLv2 is not only weak, but actively harmful to the TLS ecosystem.

3 Introduction

TLS [13] is one of the main protocols responsible for transport security on the modern Internet. TLS and its precursor SSLv3 have been the target of a large number of cryptographic attacks in the research community, both on popular implementations and the protocol itself [33]. Prominent recent examples include attacks on outdated or deliberately weakened encryption in RC4 [3], RSA [5], and Diffie-Hellman [1], different side channels including Lucky13 [2], BEAST [14], and POODLE [35], and several attacks on invalid TLS protocol flows [5, 6, 12]. Comparatively little attention has been paid to the SSLv2 protocol, likely because the known attacks are so devastating and the protocol has long been considered obsolete. Wagner and Schneier wrote in 1996 that their attacks on SSLv2 "will be irrelevant in the long term when servers stop accepting SSL 2.0 connections" [41]. Most modern TLS clients do not support SSLv2 at all. Yet in 2016, our Internet-wide scans find that out of 36 million HTTPS servers, 6 million (17%) support SSLv2. A Bleichenbacher attack on SSLv2. Bleichenbacher's padding oracle attack [8] is an adaptive chosen ciphertext attack against PKCS1 v1.5, the RSA padding standard used in SSL and TLS. It enables decryption of RSA ciphertexts if a server distinguishes between correctly and incorrectly

padded RSA plaintexts, and was termed the "million-message attack" upon its introduction in 1998, after the number of decryption queries needed to deduce a plaintext. All widely used SSL/TLS servers include countermeasures against Bleichenbacher attacks. Our first result shows that the SSLv2 protocol is fatally vulnerable to a form of Bleichenbacher attack that enables decryption of RSA ciphertexts. We develop a novel application of the attack that allows us to use a server that supports SSLv2 as an efficient padding oracle. This attack is a protocol-level flaw in SSLv2 that results in a feasible attack for 40-bit export cipher strengths, and in fact abuses the universally implemented countermeasures against Bleichenbacher attacks to obtain a decryption oracle. We also discovered multiple implementation flaws in commonly deployed OpenSSL versions that allow an extremely efficient instantiation of this attack. Using SSLv2 to break TLS. Second, we present a novel cross-protocol attack that allows an attacker to break a passively collected RSA key exchange for any TLS server if the RSA keys are also used for SSLv2, possibly on a different server. We call this attack DROWN (Decrypting RSA using Obsolete and Weakened eNcryption). In its general version, the attack exploits the protocol flaws in SSLv2, does not rely on any particular library implementation, and is feasible to carry out in practice by taking advantage of commonly supported export-grade ciphers. In order to decrypt one TLS session, the attacker must passively capture about 1,000 TLS sessions using RSA key exchange, make 40,000 SSLv2 connections to the victim server, and perform 2 50 symmetric encryption operations. We successfully carried out this attack using an optimized GPU implementation and were able to decrypt a 2048-bit RSA ciphertext in less than 18 hours on a GPU cluster and less than 8 hours using Amazon EC2. We found that 11.5 million HTTPS servers (33%) are vulnerable to this attack, because many HTTPS servers that do not directly support SSLv2 share RSA keys with other services that do. Of servers offering HTTPS with browser-trusted certificates, 22% are vulnerable. We also present a special version of DROWN that exploits flaws in OpenSSL for a more efficient oracle. It requires roughly the same number of captured TLS sessions as the general attack, but only half as many connections to the victim server and no large computations. This attack can be completed on a single core on commodity hardware in less than a minute, and is limited primarily by how fast the server can complete handshakes. It is fast enough that an attacker can perform man-in-the-middle attacks on live TLS sessions before the handshake times out, and downgrade a modern TLS client to RSA key exchange with a server that prefers non-RSA cipher suites. Our Internet-wide scans suggest that 79% of HTTPS servers that are vulnerable to the general attack, or 26% of all HTTPS servers, are also vulnerable to real-time attacks exploiting these implementation flaws. Our results highlight the risk that continued support for SSLv2 imposes on the security of much more recent TLS versions. This is an instance of a more general phenomenon of insufficient domain separation, where older, vulnerable security standards can open the door to attacks on newer versions. We conclude that phasing out outdated and insecure standards should become a priority for standards designers and practitioners. Disclosure. DROWN was assigned CVE-2016-0800. We disclosed our attacks to OpenSSL and worked

with them to coordinate further disclosures. The specific OpenSSL vulnerabilities we discovered have been designated CVE-2015-3197, CVE-2016-0703, and CVE-2016- 0704. In response to our findings, OpenSSL has made it impossible to configure a TLS server in such a way that it is vulnerable to DROWN. Microsoft had already disabled SSLv2 for all supported versions of IIS. We also disclosed the attack to the NSS developers, who have disabled SSLv2 on the last NSS tool that supported it and have hastened efforts to entirely remove the protocol from their codebase. In response to our disclosure, Google will disable QUIC support for non-whitelisted servers and modify the QUIC standard. We also notified IBM, Cisco, Amazon, the German CERT-Bund, and the Israeli CERT. Online resources. Contact information, server test tools, and updates are available at https://drownattack.com.