CZ4062 Computer Security Case Study

Group 88
Parashar Kshitij
Dhanyamraju Harsh Rao
Malavade Sanskar Deepak
Lek Wei Chen

01 Pegasus

case study on iOS

01

Introduction

What is Pegasus Spyware?

02

Spyware

How are the attacks performed?

03

Device Infection

What are its capabilities?

04

iOS vulnerabilities

iOS kernel and Webkit vulnerabilities

05

Persistence and Secrecy

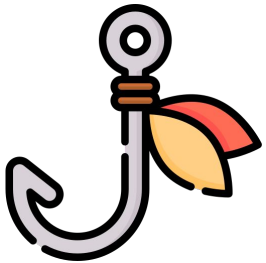
How the spyware hides itself

06

Data Gathering

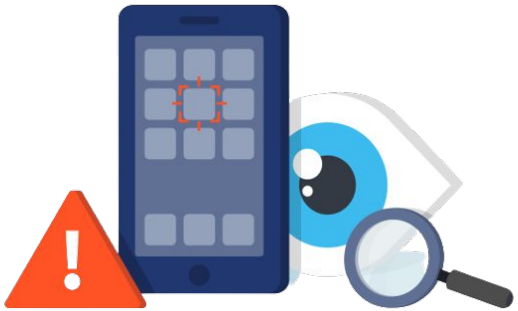
How data are being gathered

Brief Introduction



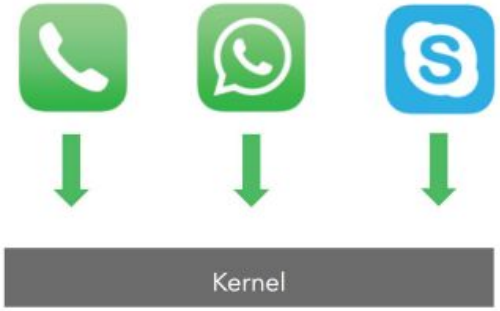
"Hooks" onto victim's devices

Remotely accessing data information from applications.

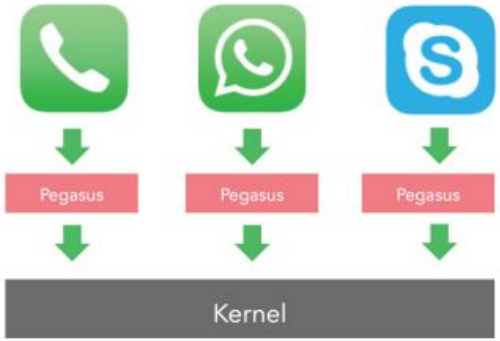


Steals classified information for further exploitation.

Normal Phone



Pegasus
Infected Phone



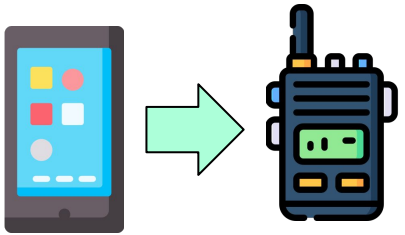
Simple illustration of how pegasus function

Audio, text data are intercepted by pegasus in real-time before being sent into kernel.

Spyware

Jailbreaks User's Phone

Compromises originally installed Apps



Attacker can access: Call Logs, SMS, Audio and Video Communications

"It (Pegasus) turns the Phone into a Walkie Talkie"

- Co-Founder, NSO Group

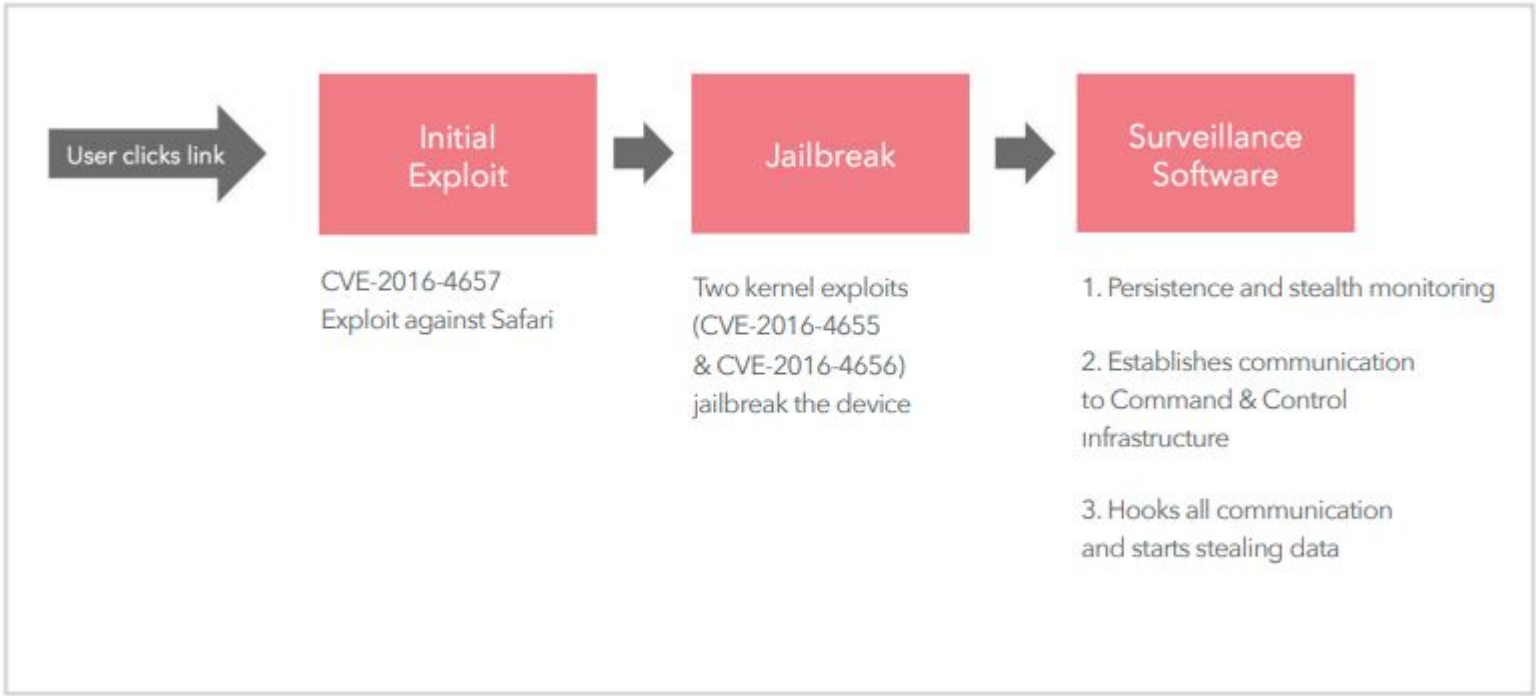
Device Infection

Malicious Website URL sent to Identified Target via SMS

Website exploits the web browser on the target's device and any other components

Zero-click vector: A special type of SMS message called WAP push service loading message, automatically opens the link in a web browser instance.

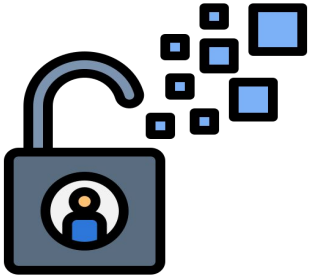
Attack Sequence



Trident vulnerabilities

In iOS 9.3.4 (later patched in iOS 9.3.5)

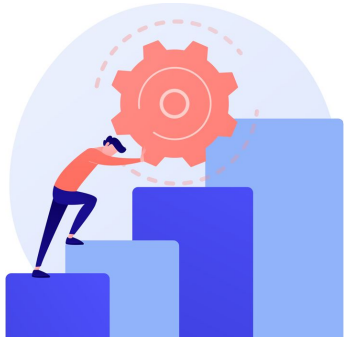
CVE-2016-4657
Memory Corruption in Safari WebKit allows an attacker to execute arbitrary code. Used to obtain code execution privileges



CVE-2016-4655
Kernel Information Leak Circumvents KASLR leaks a non-obfuscated kernel memory address in the return value. Used to access Kernel

CVE-2016-4656
Memory Corruption leads to Jailbreak
This vulnerability is used to jailbreak the device, using a memory corruption vulnerability in the kernel

Persistence and Secrecy



Implements its persistence mechanism through a memory corruption issue in a developer tool called “jsc” allowing users to execute javascript using the WebKit engine outside the context of a web browser.

Constantly monitors the phone for status and disables any other access to the phone by previous/other jailbreaking software. Also has self destruct tool to wipe itself from the device.

Persistence and Secrecy

Disabling Updates
The Stage 3 loader ensures that the phone wont receive auto-updates going forward

JailBreak Detection
The Stage 3 loader also checks the device to see if it had been previously jailbroken

```
BOOL is_jail()
{
 return (unsigned __int8)is_file_exist((int)CFSTR("/pguntether"))
     || (unsigned
__int8)is_file_exist((int)CFSTR("/System/Library/LaunchDaemons/com.saurik.Cydi
a.Startup.plist"));
}
```

Data Gathering

Gathers everything from obvious high-value data like passwords, contacts, and calendar entries to data from numerous social networks.

Example: The software gathers contacts from the system dumping the victim's entire address book.

```
v3 = CFSTR("/private/var/mobile/Library/AddressBook/AddressBook.sqlitedb");
v4 =
CFSTR("/private/var/mobile/Library/AddressBook/AddressBookImages.sqlitedb");


@property (nonatomic) unsigned int m6cVniVZHP7fjJGSl;
@property (retain, nonatomic) NSString *n7UaDOxao5xVD;
@property (retain, nonatomic) NSString *namePrefix;
@property (retain, nonatomic) NSString *firstName;
@property (retain, nonatomic) NSString *middleName;
@property (retain, nonatomic) NSString *lastName;
@property (retain, nonatomic) NSString *nameSuffix;
@property (retain, nonatomic) NSString *nickname;
@property (retain, nonatomic) NSString *organization;
@property (retain, nonatomic) NSString *department;
@property (retain, nonatomic) NSString *title;
@property (retain, nonatomic) NSString *h4fW1CC56Q;
@property (retain, nonatomic) NSData *imageData;
@property (retain, nonatomic) NSDate *birthday;
@property (readonly) s62tW6JOsHqCefoKFMkoTgOHc *emails;
@property (readonly) s62tW6JOsHqCefoKFMkoTgOHc *phones;
@property (readonly) s62tW6JOsHqCefoKFMkoTgOHc *addresses;
```

Impacts of Pegasus

OCCRP ORGANIZED CRIME AND CORRUPTION REPORTING PROJECT

Pegasus Scandal Hits EU from Within

Published: 15 November 2022 WRITTEN BY ZDRAVKO LJUBAS

Pegasus scandal: the dismissal of a Moroccan civil servant rocks UNESCO

NDTV LIVE TV LATEST INDIA ELECTIONS OPINION VIDEO CITIES

Home > India News > "Pegasus (Spyware) Has My Number": Prashant Kishor's Swipe At BJP

"Pegasus (Spyware) Has My Number": Prashant Kishor's Swipe At BJP

Prashant Kishor made a reference to the spyware while debunking the accusation touch, clandestinely, with Bihar Chief Minister Nitish Kumar.

ews | Press Trust of India | Updated: November 01, 2022 11:17 pm IST

Prevention Steps

Don’t Click on random links

Patch Consistently

Ensure Device Security undergo update regularly

02

HeartBleed

CVE - 2014 - 0160

01

Introduction

Basic Introduction

02

Transport Layer Security

Internet Security Protocol

03

Heartbeat

Checking if connection is active

04

Heartbleed Exploit

The code

05

Impact

Companies Affected & Data Leaked

06

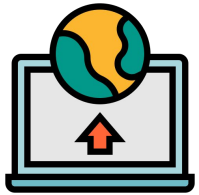
Patch

Bug Fixes

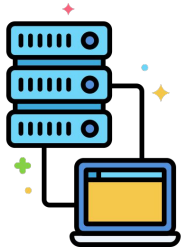
Introduction

A security bug in the OpenSSL cryptography library version 1.0.1

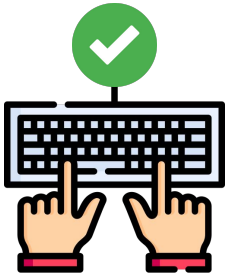
OpenSSL is used in implementation of the Transport Layer Security (**TLS**) protocol.



Introduced into the software in 2012 and publicly disclosed in April 2014



Could be exploited regardless of whether the vulnerable OpenSSL instance is running as a TLS server or client.

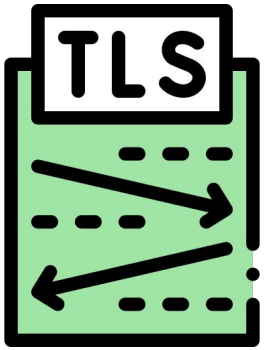


It resulted from improper input validation (due to a missing bounds check) in the implementation of the TLS heartbeat extension.

Transport Layer Security (TLS)

Security Protocol

Encrypting Communication between client and servers

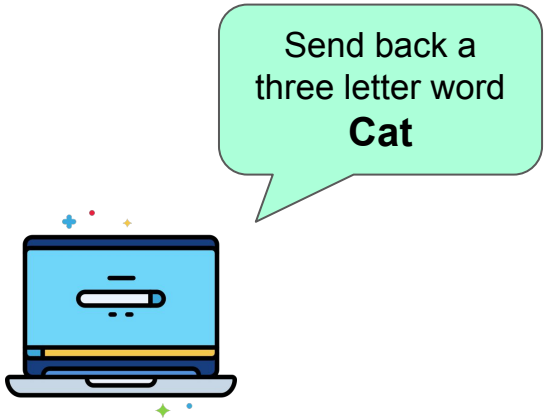


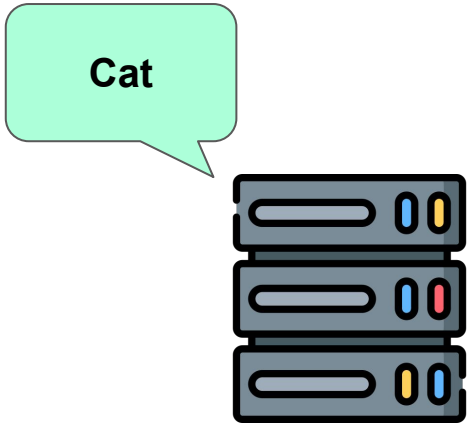
Connection initiated by sequence known as TLS Handshake

HeartBeat

Ensure Connection is Active

Not Encrypted



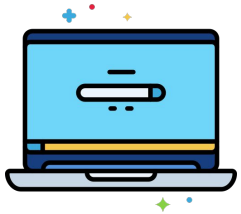


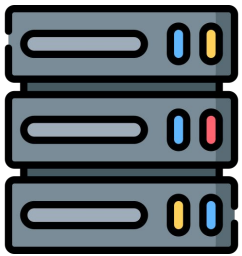
HeartBleed Exploit

Buffer OverRead

Send back a eleven letter word **Cat**

Cat12345678





HeartBleed Exploit Code

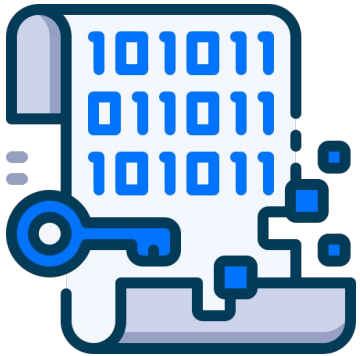
```c
/* This is simplified version of the source code */
int dtls1_process_heartbeat(SSL *s)
{
    // Define variables
    unsigned short hbtype; unsigned int payload_len;
    unsigned int padding = 16;
    unsigned char *p = &s->s3->rrec.data[0];
    // Read type and length of payload
    hbtype = *p++;
    n2s(p, payload_len);  // Call by reference
    ...
    // Check request type
    if (hbtype == TLS1_HB_REQUEST)
    {
        unsigned char *buffer, *bp; int r;
        // Allocate Memory for response
        buffer = OPENSSL_malloc(1 + 2 + payload_len + padding);
        bp = buffer;

        *bp++ = TLS1_HB_RESPONSE;
        s2n(payload_len, bp);
        memcpy(bp, pl, payload_len);
        bp += payload_len;
        RAND_pseudo_bytes(bp, padding);

        // Reformat & Send response back
        ...
    }
    ...
}
```

Output Format:

Message Type	PayLoad Length	PayLoad Content	Padding
1 byte	2 bytes		16 bytes

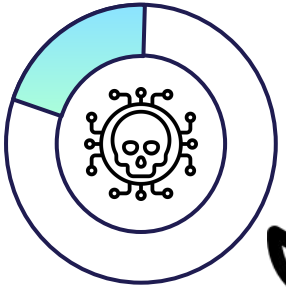


Impact

17%

SSL Web Server

Not all SSL Servers supported Heart Beat



Eventbrite

Canada revenue agency

coles

yahoo!

imgur

CHS

okc

mumsnet
By parents for parents

Personal Details
Emails
Passwords
Private Encryption Key
Credit Card Numbers
Medical Records
Social Media Chats

2012
Introduced

April 2014
Discovered

Same Day
Patched

HeartBleed Exploit Patch

```c
/* This is simplified version of the source code */
int dtls1_process_heartbeat(SSL *s)
{
    // Define variables
    unsigned short hbtype; unsigned int payload_len;
    unsigned int padding = 16;
    unsigned char *p = &s->s3->rrec.data[0];
    // Read type and length of payload
    hbtype = *p++;
    n2s(p, payload_len);  // Call by reference
    pl = p;
    ...
    // Check request type
    if (hbtype == TLS1_HB_REQUEST)
    {
        unsigned char *buffer, *bp; int r;
        // Allocate Memory for response
        buffer = OPENSSL_malloc(1 + 2 + payload_len + padding);
        bp = buffer;

        *bp++ = TLS1_HB_RESPONSE;
        s2n(payload_len, bp);
        memcpy(bp, pl, payload_len);
        bp += payload_len;
        RAND_pseudo_bytes(bp, padding);

        // Reformat & Send response back
        ...
    }
    ...
}
```

```c
/* This is simplified version of the source code */
int dtls1_process_heartbeat(SSL *s)
{
    // Define variables
    unsigned short hbtype; unsigned int payload_len;
    unsigned int padding = 16;
    unsigned char *p = &s->s3->rrec.data[0];
    // Read type and length of payload
    hbtype = *p++;
    n2s(p, payload_len);  // Call by reference
    ...
    // Check for length (Added in patch)
    if (1 + 2 + payload_len + 16 > s->s3->rrec.length)
        return 0; /* silently discard per RFC 6520 sec. 4 */
    pl = p;
    ...
    // Check request type
    if (hbtype == TLS1_HB_REQUEST)
    {
        unsigned char *buffer, *bp; int r;
        // Allocate Memory for response
        buffer = OPENSSL_malloc(1 + 2 + payload_len + padding);
        bp = buffer;

        *bp++ = TLS1_HB_RESPONSE;
        s2n(payload_len, bp);
        memcpy(bp, pl, payload_len);
        bp += payload_len;
        RAND_pseudo_bytes(bp, padding);

        // Reformat & Send response back
        ...
    }
    ...
}
```

Encrypt Sensitive Data in Memory

References

Pegasus

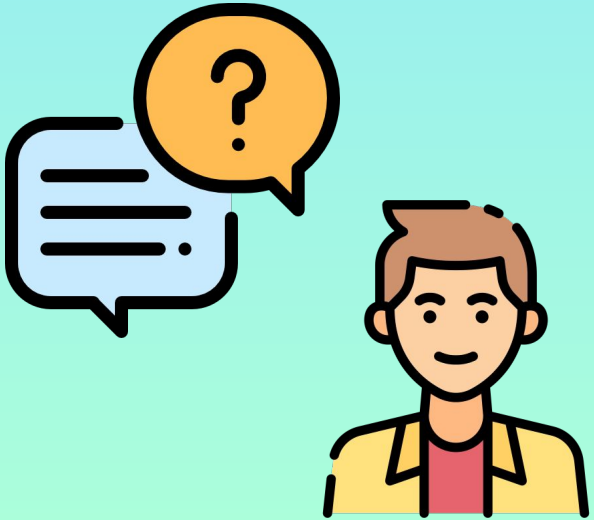
- https://techbeacon.com/security/pegasus-spyware-vulnerability-chainings-next-level
- https://info.lookout.com/rs/051-ESQ-475/images/lookout-pegasus-technical-analysis.pdf
- https://citizenlab.ca/2022/07/geckospy-pegasus-spyware-used-against-thailands-pro-democracy-movement/

Heartbleed

- https://heartbleed.com/
- https://en.wikipedia.org/wiki/Heartbleed
- https://www.cisa.gov/uscert/ncas/alerts/TA14-098A
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0160
- https://www.netcraft.com/internet-data-mining/ssl-survey/
- https://www.theguardian.com/technology/2014/apr/08/heartbleed-bug-puts-encryption-at-risk-for-hundreds-of-thousands-of-servers#:~:text=The%20bug%2C%20called%20%22Heartbleed%22,run%20affected%20versions%20of%20OpenSSL.
- https://www.bbc.com/news/technology-27028101

Thank you

Any questions?



THANKS!

Do you have any questions?
youremail@freepik.com
+91 620 421 838
yourcompany.com