Sweet QuaDreams 🌙💤 or Nightmare before Christmas? 🎄 Dissecting an iOS 0-Day

Bill Marczak, *The Citizen Lab*
Christine Fossaceca, *Microsoft*



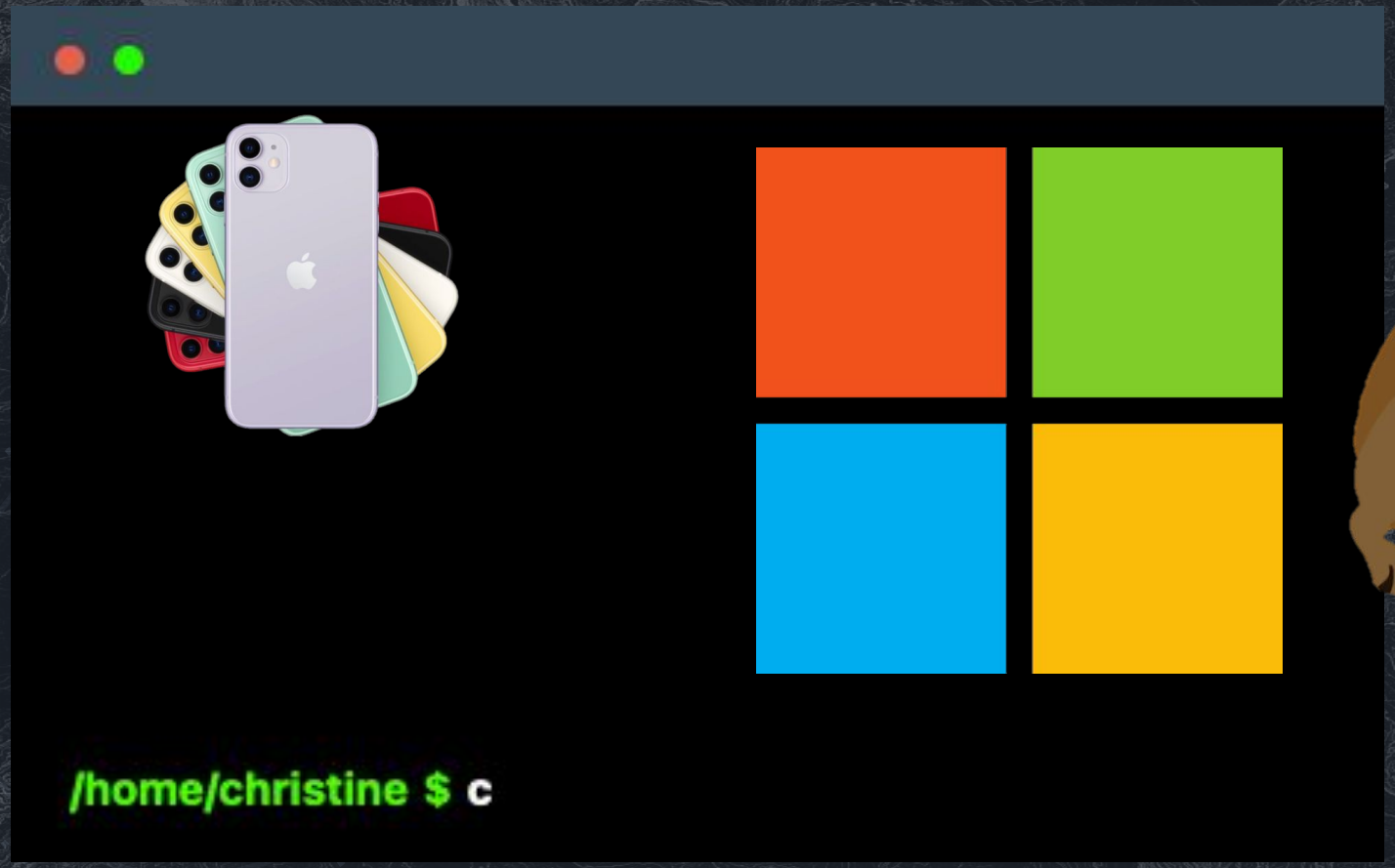




Notes:

- 🌴 We’re talking about an attack from 2021
- 🌴 We’re not dropping CVEs on stage!
- 🌴 Have shared technical details with Apple

About Christine







her hax
PODCAST

@x71n3

About Bill



About Bill



HIDE AND SEEK

Tracking NSO Group's Pegasus Spyware to Operations in 45 Countries

Running in Circles

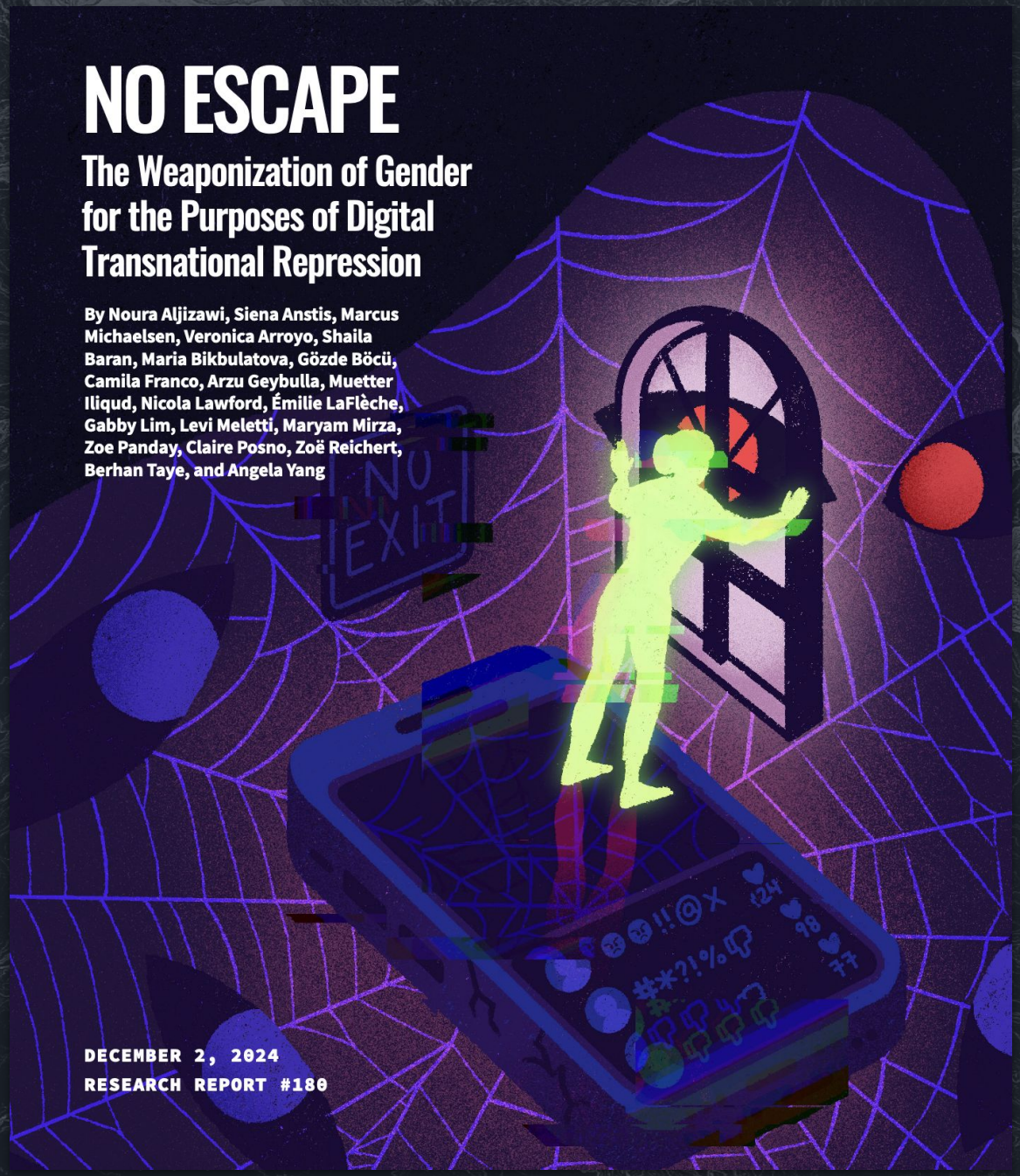
Uncovering the Clients of Cyberespionage Firm Circles

PREDATOR IN THE WIRES

Ahmed Eltantawy Targeted with Predator Spyware After Announcing Presidential Ambitions

Hooking Candiru

Another Mercenary Spyware Vendor Comes into Focus



Something to Remember Us By



Device Confiscated by Russian Authorities Returned with Monokle-Type Spyware Installed

December 5, 2024

Research

Key Findings This joint investigation with First Department, a legal assistance organization, found spyware covertly implanted on a phone returned to a Russian programmer accused of sending money to Ukraine after he was released from custody. He describes being subjected to beatings and an intense effort to recruit him as an informant for the Russian… Read more »

iPhone Initial Access

iPhone Initial Access



CVE-Whatever: Perpetual Safari/WebKit Exploit

iPhone Initial Access



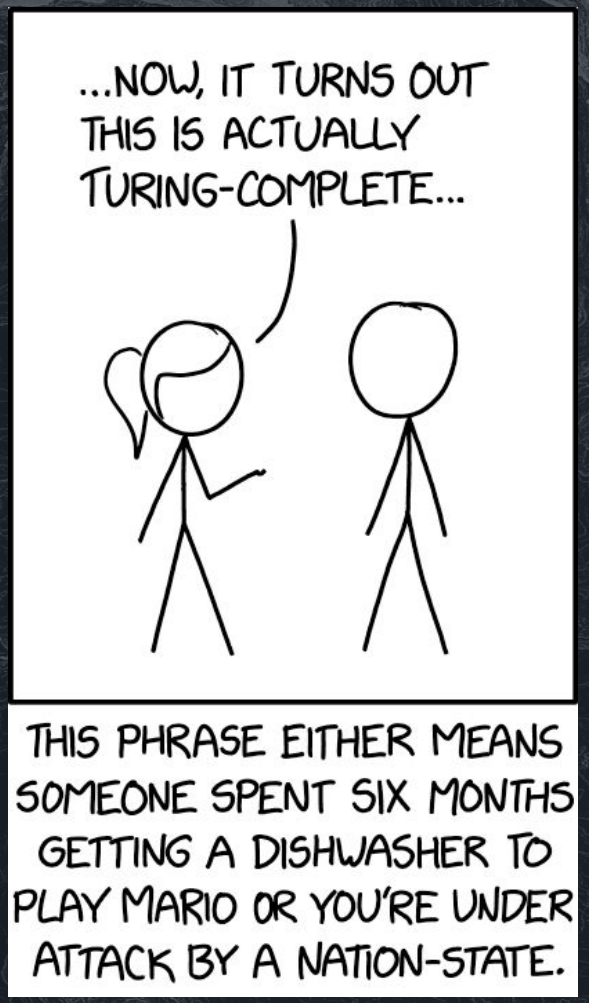
CVE-Whatever: Perpetual
Safari/WebKit Exploit



Target: Ahmed Mansoor
UAE Human Rights Activist

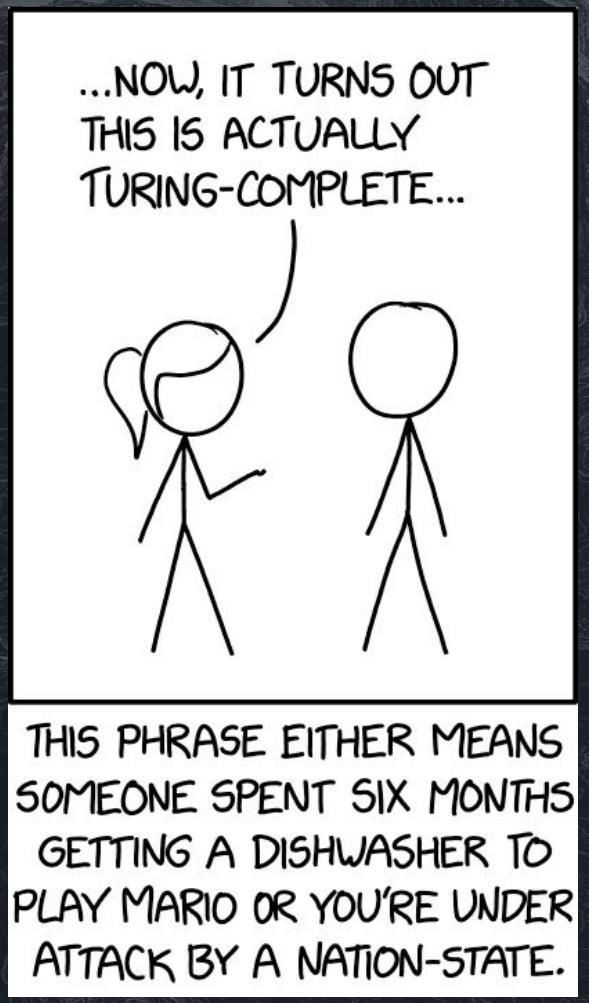
iPhone Initial Access with Zero Clicks

iPhone Initial Access with Zero Clicks



CVE-2021-30860: Integer overflow in CoreGraphics

iPhone Initial Access with Zero Clicks

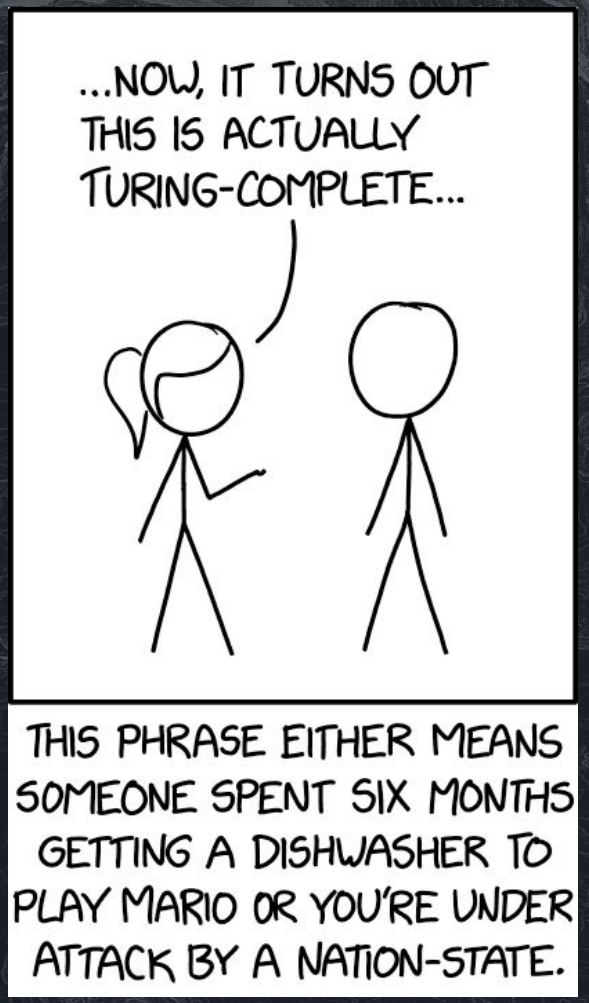


CVE-2021-30860: Integer overflow in CoreGraphics



CVE-2023-41064: Buffer overflow in ImageIO

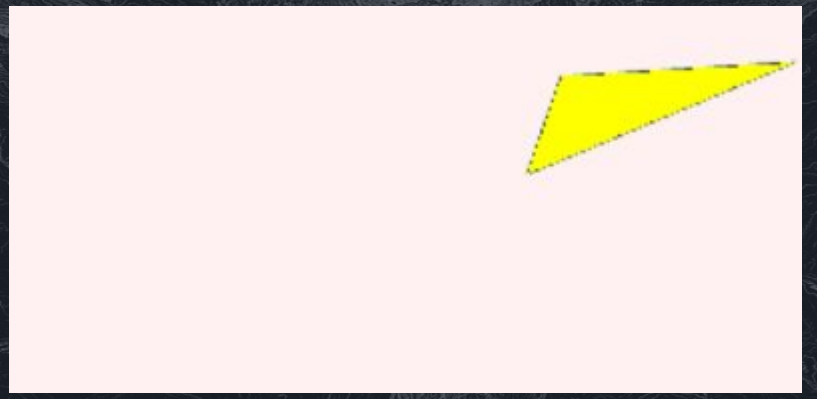
iPhone Initial Access with Zero Clicks



CVE-2021-30860: Integer overflow in CoreGraphics



CVE-2023-41064: Buffer overflow in ImageIO



CVE-2023-41990: Issue in FontParser

Our Definitions

0-day (*ze·ro·day*): an exploited vulnerability for which there is no patch available

0-click (*ze·ro·click*): a remote vulnerability that requires no user interaction (or “clicks”)

Our Definitions

0-day (*ze·ro·day*): an exploited vulnerability for which there is no patch available

0-click (*ze·ro·click*): a remote vulnerability that requires no user interaction (or “clicks”)

Our Definitions

0-day (*ze·ro·day*): an exploited vulnerability for which there is no patch available

0-click (*ze·ro·click*): a remote vulnerability that requires no user interaction (or “clicks”)

Our Definitions

0-day (*ze·ro* nerability for which there

0-click (*ze·r* erability that requires no u ks”)



Apple Sandboxes IMTranscoderAgent with BlastDoor



Apple Sandboxes IMTranscoderAgent with BlastDoor



BlastDoor: A Fork in the Road



BlastDoor: A Fork in the Road



BlastDoor: A Fork in the Road



Discovery of the Attack & Samples

Attribution: Sometimes it's Easy!

Static & Dynamic Reversing of the Sample

A Theory of the Exploit

Discovery of the Attack & Samples

Log Analysis

Log Analysis

Top-down:
Analyze a spyware sample, understand what forensic traces it leaves behind, then look for these in the phone's logs.





Log Analysis

<u>Top-down</u>:
Analyze a spyware sample, understand what forensic traces it leaves behind, then look for these in the phone's logs.



<u>Bottom-up</u>:
Look for *implausible artifacts* in the phone's logs, and then try to attribute them.
Can detect unknown spyware this way!

Examples of "Implausible Artifacts"

Examples of "Implausible Artifacts"

- Evidence that a non-iOS-update binary ran from: `/private/var/db/com.apple.xpc.roleaccountd.staging/`
- Evidence that any binary ran from `/tmp`
- Evidence that a binary consumed mobile data that is "not supposed to" (e.g., `BackupAgent`)

Examples of "Implausible Artifacts"

- Evidence that a non-iOS-update binary ran from: `/private/var/db/com.apple.xpc.roleaccountd.staging/`
- Evidence that any binary ran from `/tmp`
- Evidence that a binary consumed mobile data that is "not supposed to" (e.g., `BackupAgent`)

Mental Model: Attacker is running traditional chain (RCE -> LPE -> attacker binary). Not always true!

Examples of "Implausible Artifacts"

- Evidence that a non-iOS-update binary ran from: `/private/var/db/com.apple.xpc.roleaccountd.staging/`
- Evidence that any binary ran from `/tmp`
- Evidence that a binary consumed mobile data that is "not supposed to" (e.g., `BackupAgent`)

Mental Model: Attacker is running traditional chain (RCE -> LPE -> attacker binary). Not always true!

messaging software. “You missed it – there’s one Israeli company with hypersonic weapons, in cybersecurity terms, that leaves NSO in the dust," he replied to this reporter, “Paragon.”

An implausible artifact ITW...

Get yer' phones checked here!!!



An implausible artifact ITW...

Get yer' phones checked here!!!



Several phones showed a binary had run:
`/private/var/db/com.apple.xpc.roleaccountd.staging/subridged`

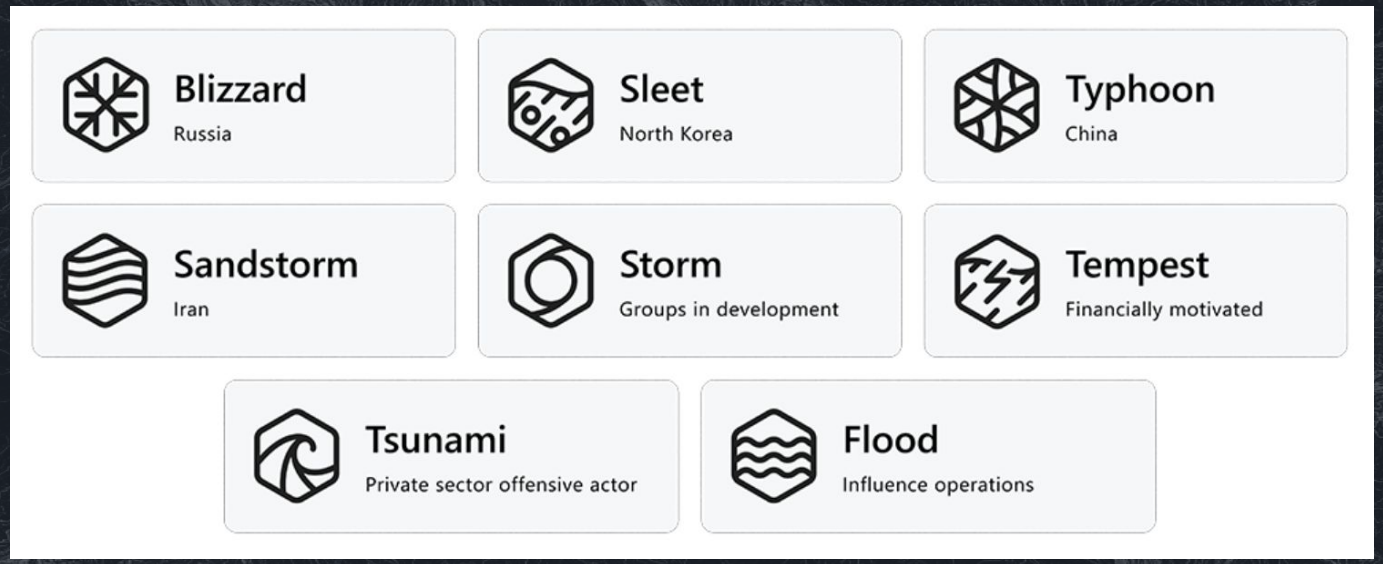
Phones negative for Pegasus!!!

...meanwhile at Microsoft

Microsoft Threat Intelligence is constantly tracking ITW threats

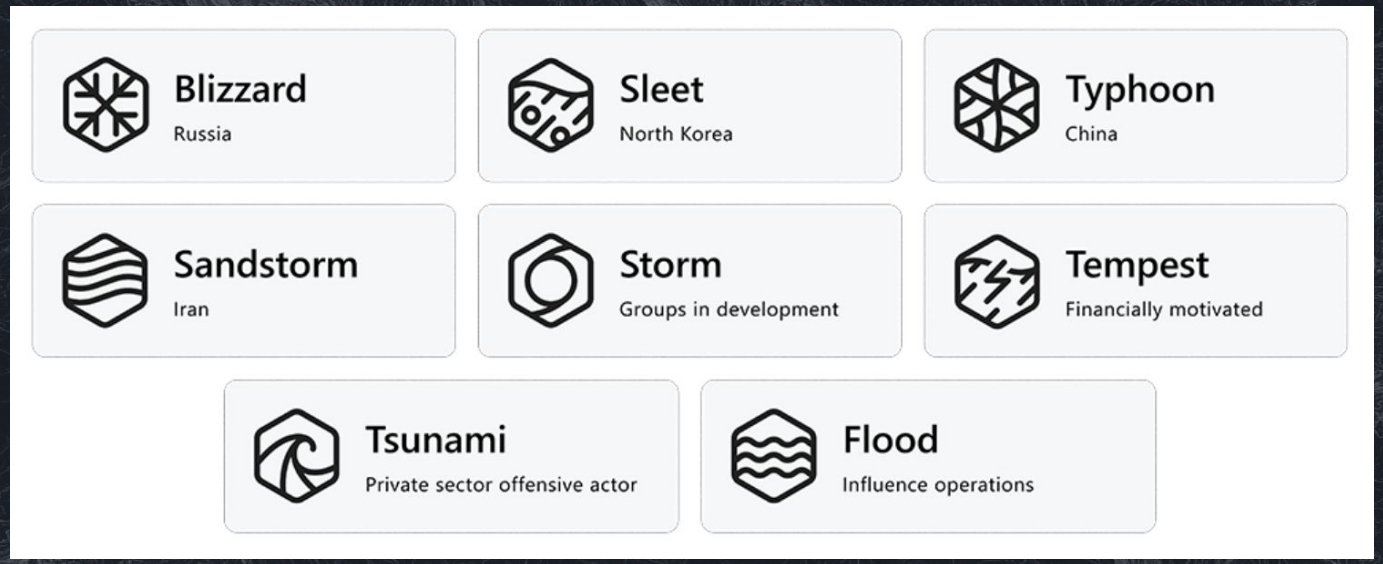
...meanwhile at Microsoft

Microsoft Threat Intelligence is constantly tracking ITW threats



...meanwhile at Microsoft

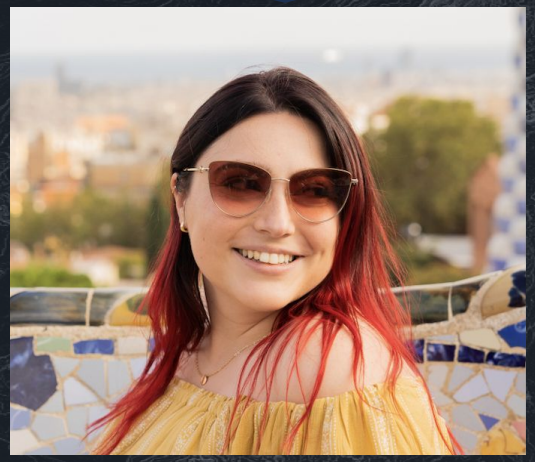
Microsoft Threat Intelligence is constantly tracking ITW threats



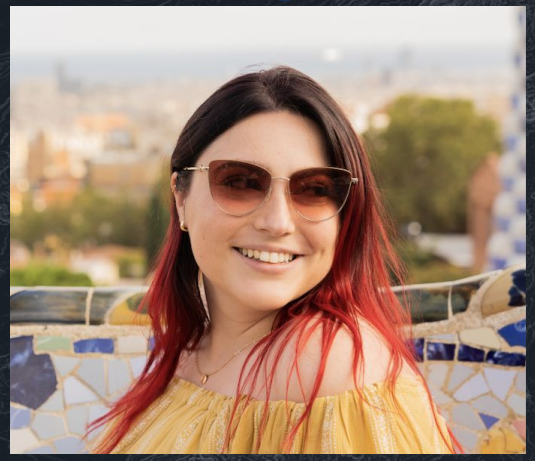
Microsoft had found a sample with this hard-coded path:

`/private/var/db/com.apple.xpc.roleaccountd.staging/subridged`

Yo Citizen Lab, we have a sample matching your IOCs...



Yo Citizen Lab, we have a sample matching your IOCs...



Tell us more...



Discovery of the Attack & Samples

Attribution: Sometimes it's Easy!

Carmine Tsunami



Carmine Tsunami

Private Sector Offensive Actor (PSOA)

- A company that sells hacking tools
- Often exclusively to governments



Carmine Tsunami

Private Sector Offensive Actor (PSOA)

- A company that sells hacking tools
- Often exclusively to governments

In this case, QuaDream!



The Mercenary Spyware Industry

The Mercenary Spyware Industry

NSO GROUP

intellexa

Protect
PROTECT ELECTRONIC SYSTEMS LLC
بروتكت للأنظمة الالكترونية ذ.م.م

PARAGON



The Industry in the News

The Industry in the News

Israeli spyware used ‘extensively’ on separatists in Spain, group says

Pegasus phone spyware used to target 30 Thai activists, cyber watchdogs say

Pegasus spyware used in ‘jaw-dropping’ phone hacks on El Salvador journalists

Mexico: reporters and activists hacked with NSO spyware despite assurances

More Polish opposition figures found to have been targeted by Pegasus spyware

The Industry in the News

Dubai ruler hacked ex-wife using NSO Pegasus spyware, high court judge finds

Sheikh Mohammed used spyware on Princess Haya and five associates in unlawful abuse of power, judge rules

- 'The walls are closing in on me': the hacking of Princess Haya
- Ruling in Princess Haya case raises fresh questions for Cherie Blair



Other Mobile Spyware (0-click)



RUSSIA'S FSB HAS UNCOVERED AN INTELLIGENCE OPERATION BY US INTELLIGENCE AGENCIES USING APPLE MOBILE DEVICES

DARKMATTER
GUARDED BY GENIUS

INSIDE THE UAE'S SECRET HACKING TEAM OF AMERICAN MERCENARIES

QUADREAM: SOME SUSPECTED OPERATOR LOCATIONS

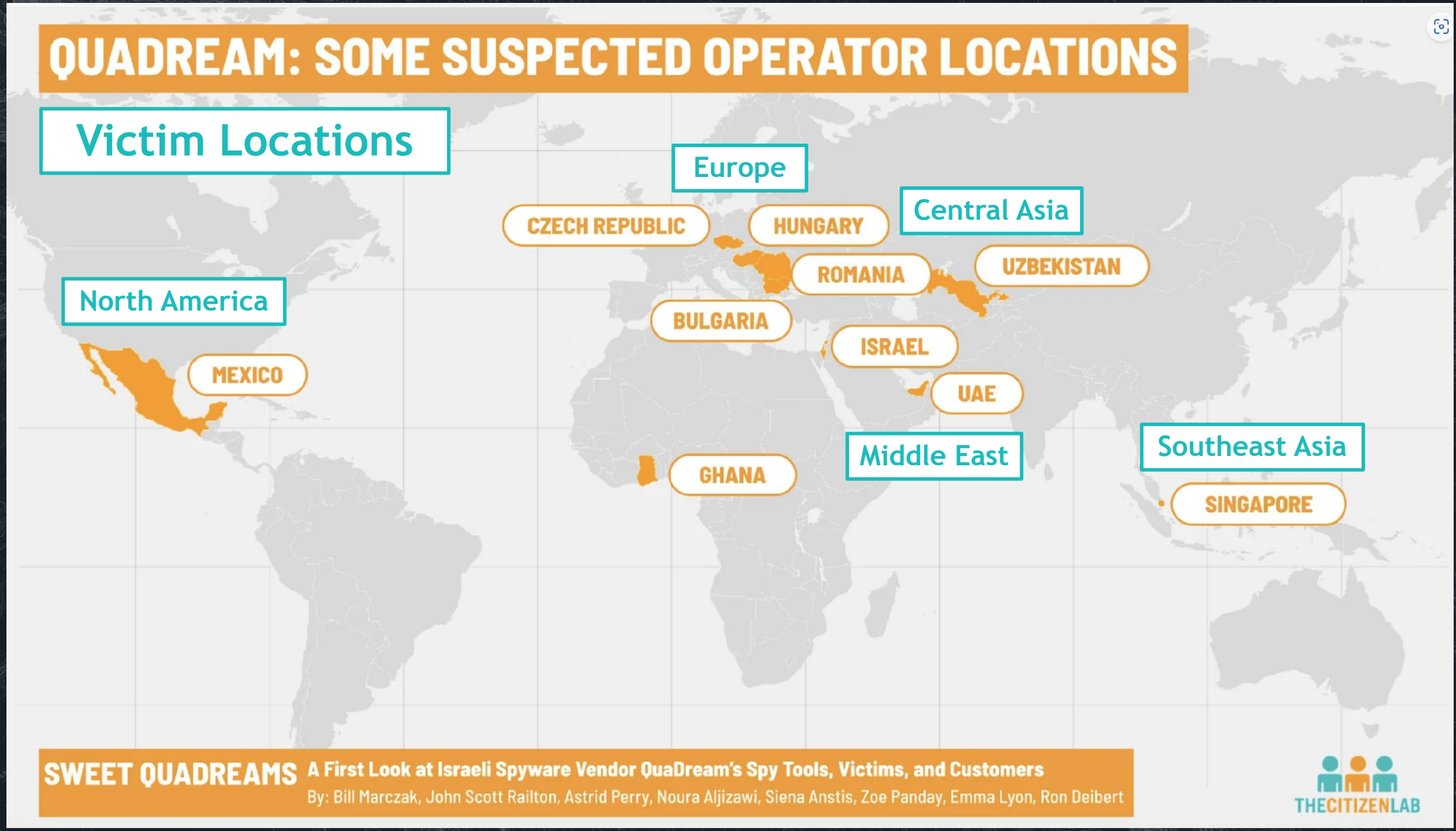


SWEET QUADREAMS A First Look at Israeli Spyware Vendor QuaDream's Spy Tools, Victims, and Customers

By: Bill Marczak, John Scott Railton, Astrid Perry, Noura Aljizawi, Siena Anstis, Zoe Panday, Emma Lyon, Ron Deibert

THECITIZENLAB

QUADREAM: SOME SUSPECTED OPERATOR LOCATIONS



SWEET QUADREAMS A First Look at Israeli Spyware Vendor QuaDream's Spy Tools, Victims, and Customers

By: Bill Marczak, John Scott Railton, Astrid Perry, Noura Aljizawi, Siena Anstis, Zoe Panday, Emma Lyon, Ron Deibert

THECITIZENLAB

UK-based dissident can sue Saudi Arabia for alleged spyware, court rules

By **Reuters**

October 21, 2024 5:03 AM PDT · Updated a month ago



Yahya Assiri

UK-based dissident can sue Saudi Arabia for alleged spyware, court rules

By **Reuters**

October 21, 2024 5:03 AM PDT · Updated a month ago



Yahya Assiri

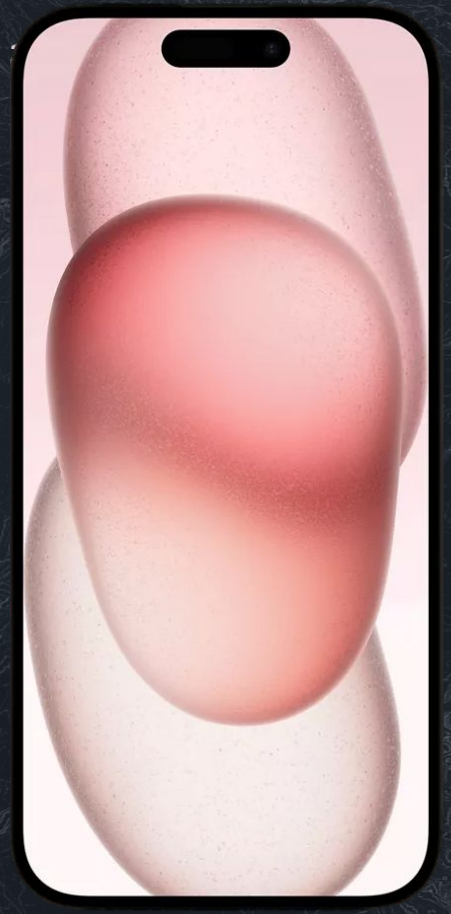
He is suing Saudi Arabia at London's High Court, saying the country used Pegasus - made by Israeli company NSO Group and sold only to nation states - and other spyware made by lesser-known Israeli firm QuaDream because of his work with dissidents.

Discovery of the Attack & Samples

Attribution: Sometimes it's Easy!

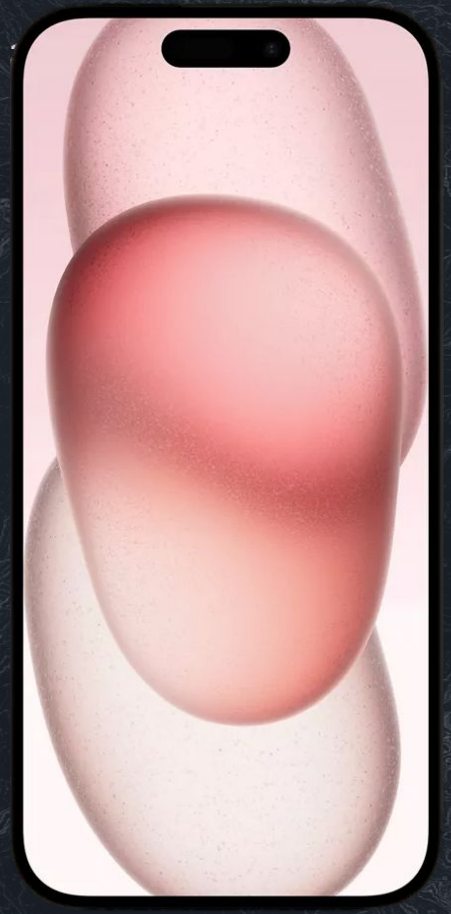
Static & Dynamic Reversing of the Sample

iOS System Protections



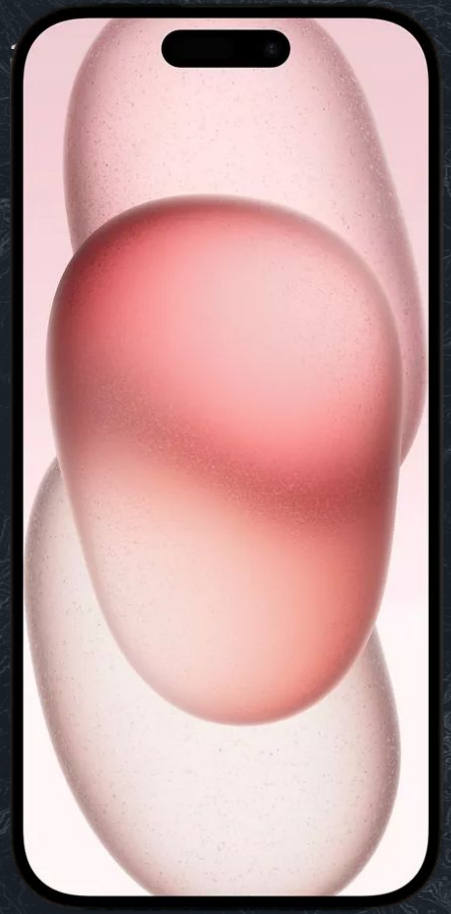
Protection Mechanism	Bypassed?
ASLR and NX	✅
Sandboxing	✅
Entitlements	✅
Codesigning + AMFI	✅
PAC	✅
PPL	✅

iOS System Protections



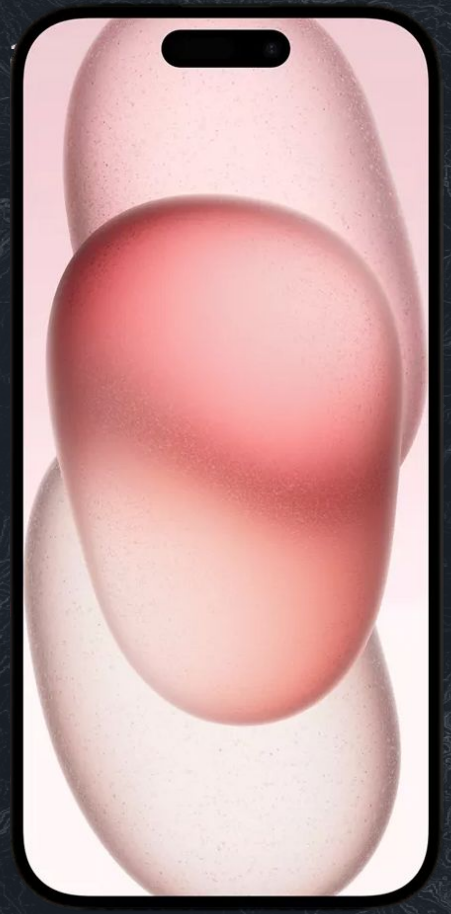
Protection Mechanism	Bypassed?
ASLR and NX	✓
Sandboxing	✓
Entitlements	✓
Codesigning + AMFI	✓
PAC	✓
PPL	✓

iOS System Protections



Protection Mechanism	Bypassed?
ASLR and NX	✓
Sandboxing	✓
Entitlements	✓
Codesigning + AMFI	✓
PAC	✓
PPL	✓

iOS System Protections



Protection Mechanism	Bypassed?
ASLR and NX	✓
Sandboxing	✓
Entitlements	✓
Codesigning + AMFI	✓
PAC	✓
PPL	✓

Sample Capabilities

Sample Capabilities

- Device Info
  - Wi-Fi
  - Airplane Mode
  - Carrier Info
  - iOS version



Sample Capabilities

- Device Info
  - Wi-Fi
  - Airplane Mode
  - Carrier Info
  - iOS version

- Spying
  - Records audio
  - Takes pictures
  - Tracks location













Sample Capabilities

Sample Capabilities

- Exfiltrates and deletes keychain items



Sample Capabilities

- Exfiltrates and deletes keychain items
- Exfiltrates and deletes other files on disk





Sample Capabilities

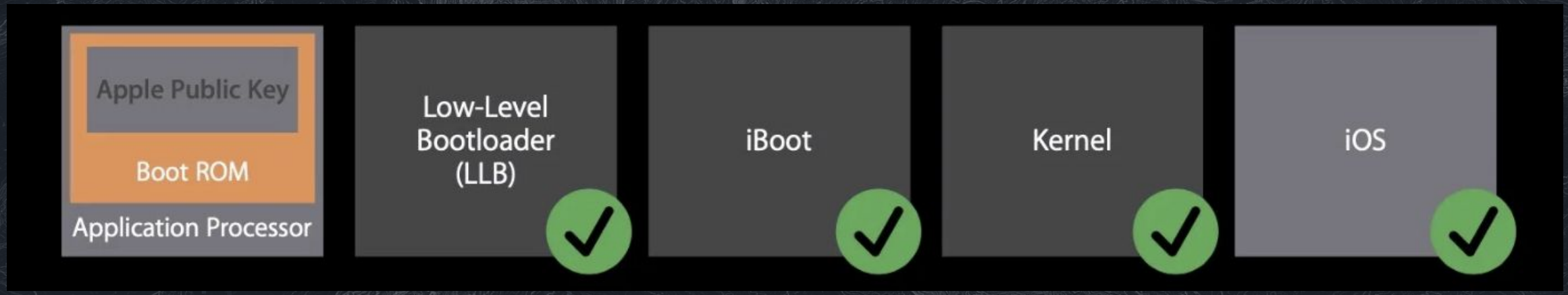
- Exfiltrates and deletes keychain items
- Exfiltrates and deletes other files on disk
- NO persistence mechanism!







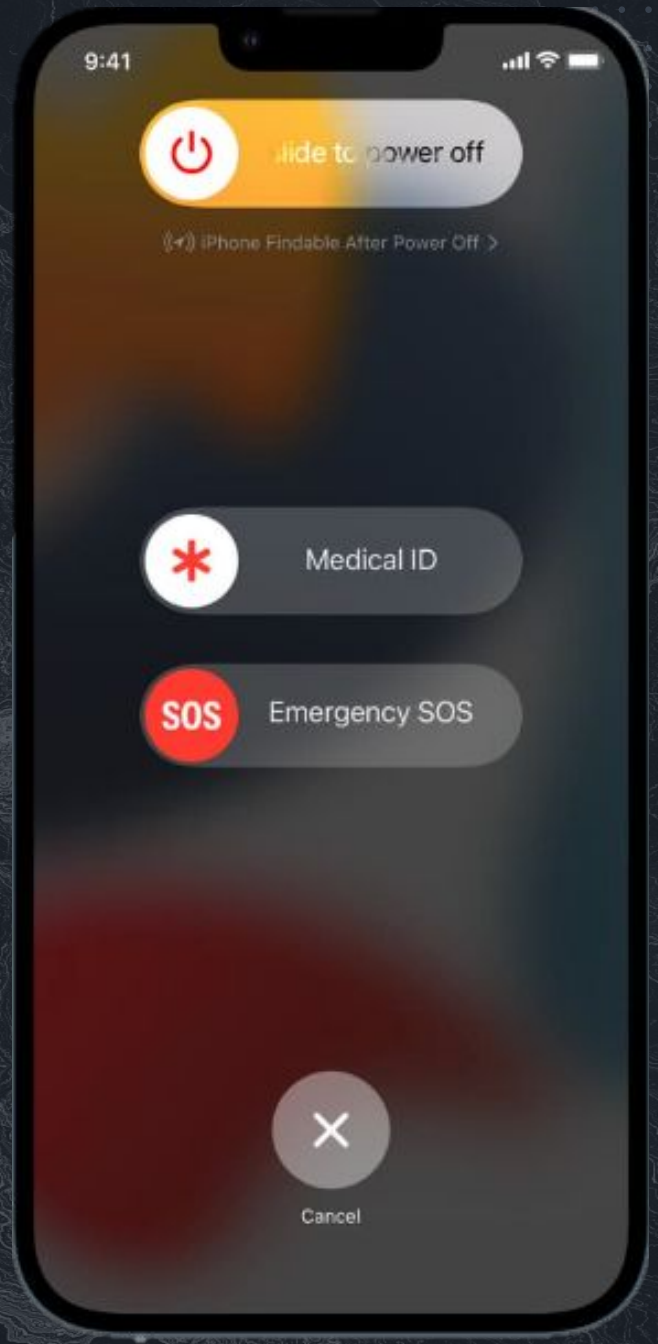
iOS Secure Boot Chain



Apple WWDC 2016

Examples of iOS “persistence”

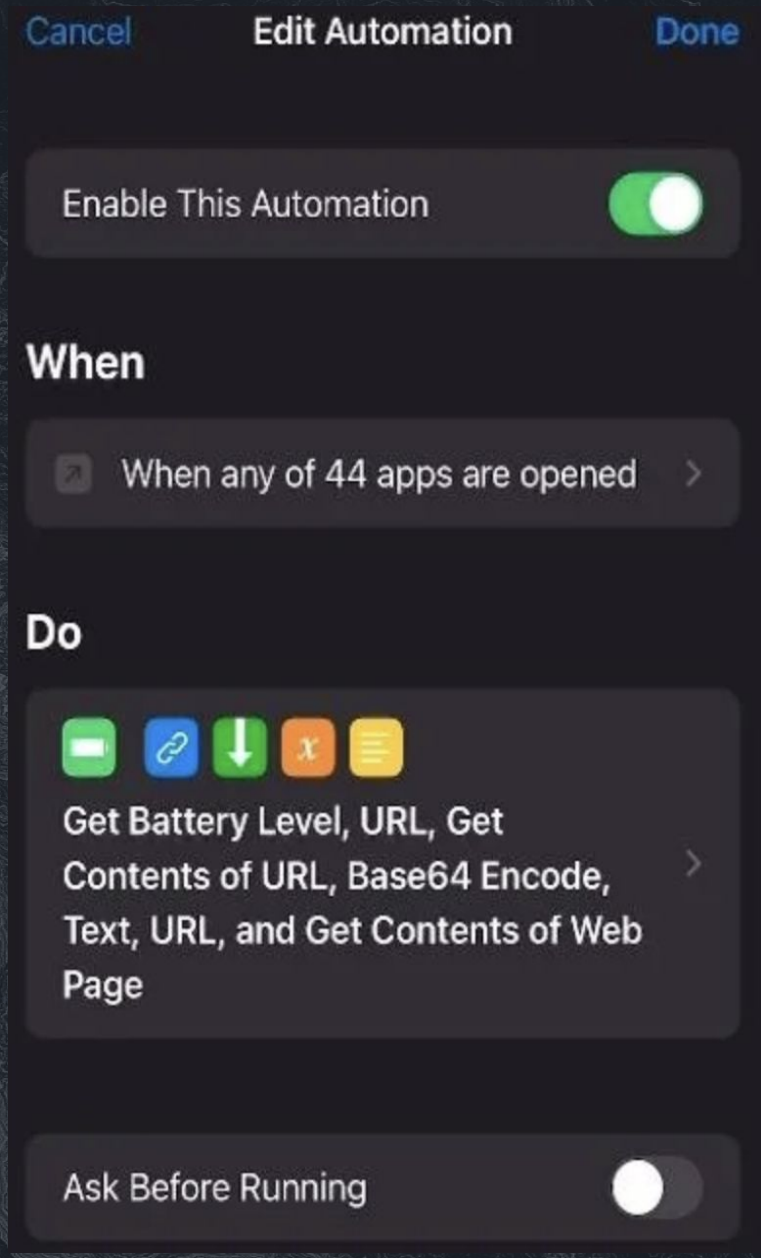
- Zecops Blog: “NoReboot”. Hook shutdown mechanism to "fake" a reboot
  (theoretical attack - not ITW)



Examples of iOS “persistence”

- Zecops Blog: “NoReboot”. Hook shutdown mechanism to "fake" a reboot
  (theoretical attack - not ITW)

- Re-infect on reboot examples:
  - Pegasus in 2016: rtbuddyd --early-boot. Replace *rtbuddyd* w/ *JSC*, put JS exploit in file called "*--early-boot*"
  - Predator in 2021: iOS shortcut automations
  - Deserialization exploits



Examples of iOS “persistence”

- Zecops Blog: “NoReboot”. Hook shutdown mechanism to "fake" a reboot
  (theoretical attack - not ITW)

- Re-infect on reboot exampl
  - Pegasus in 20
    R
    …early-boot"
  - Predator in 2021: iOS shortcut automations

  - Deserialization exploits



Persistence is a misfeature with 0-click

Subverting iCloud 2FA

- */usr/libexec/adid* (Anisette) process is responsible
- This is “hard” to reverse (FairPlay DRM)
- So, they treat it like a black box!
- Dylib injection to inject code into *adid*, then function hooking to generate codes
- How does this work?

Dylib Injection

```
➜  OBTS 🌴
//lib injection code (thanks newosxbook.com)

//grab the task port for the target pid
task_t remoteTask;
task_for_pid(mach_task_self(), pid, &remoteTask);

//allocate memory
mach_vm_allocate( remoteTask, &remoteMem64, MEM_SIZE, VM_FLAGS_ANYWHERE);

//write shellcode into memory
mach_vm_write(remoteTask, remoteMem64, ptr_to_shellcode,len);

//make memory executable
vm_protect(remoteTask, remoteMem64, SIZE, FALSE,VM_PROT_READ|VM_PROT_EXECUTE);
```

Dylib Injection

```
➜  OBTS 🌴
//lib injection code (thanks newosxbook.com)

//grab the task port for the target pid
task_t remoteTask;
task_for_pid(mach_task_self(), pid, &remoteTask);

//allocate memory
mach_vm_allocate( remoteTask, &remoteMem64, MEM_SIZE, VM_FLAGS_ANYWHERE);

//write shellcode into memory
mach_vm_write(remoteTask, remoteMem64, ptr_to_shellcode,len);

//make memory executable
vm_protect(remoteTask, remoteMem64, SIZE, FALSE,VM_PROT_READ|VM_PROT_EXECUTE);
```

Find pid of adid

Dylib Injection

```
➜  OBTS 🌴
//lib injection code (thanks newosxbook.com)

//grab the task port for the target pid
task_t remoteTask;
task_for_pid(mach_task_self(), pid, &remoteTask);

//allocate memory
mach_vm_allocate( remoteTask, &remoteMem64, MEM_SIZE, VM_FLAGS_ANYWHERE);

//write shellcode into memory
mach_vm_write(remoteTask, remoteMem64, ptr_to_shellcode,len);

//make memory executable
vm_protect(remoteTask, remoteMem64, SIZE, FALSE,VM_PROT_READ|VM_PROT_EXECUTE);
```

allocate memory

Dylib Injection

```
➜  OBTS 🌴
//lib injection code (thanks newosxbook.com)

//grab the task port for the target pid
task_t remoteTask;
task_for_pid(mach_task_self(), pid, &remoteTask);

//allocate memory
mach_vm_allocate( remoteTask, &remoteMem64, MEM_SIZE, VM_FLAGS_ANYWHERE);

//write shellcode into memory
mach_vm_write(remoteTask, remoteMem64, ptr_to_shellcode,len);

//make memory executable
vm_protect(remoteTask, remoteMem64, SIZE, FALSE,VM_PROT_READ|VM_PROT_EXECUTE);
```

write shellcode

Dylib Injection

```
➜  OBTS 🌴
//lib injection code (thanks newosxbook.com)

//grab the task port for the target pid
task_t remoteTask;
task_for_pid(mach_task_self(), pid, &remoteTask);

//allocate memory
mach_vm_allocate( remoteTask, &remoteMem64, MEM_SIZE, VM_FLAGS_ANYWHERE);

//write shellcode into memory
mach_vm_write(remoteTask, remoteMem64, ptr_to_shellcode,len);

//make memory executable
vm_protect(remoteTask, remoteMem64, SIZE, FALSE,VM_PROT_READ|VM_PROT_EXECUTE);
```

make executable

Dylib Injection

```
➜  OBTS 🌴
//lib injection code continued

//shellcode contains dlopen pointer callback
uint64_t addrOfDlopen = (uint64_t) dlopen;

//dylib is on disk
*path_to_dylib = "/path/to/mydylib"


//when remote thread executes

callBackFunction(*addrOfDlopen, *path_to_dylib)
```

Dylib Injection

```
➜  OBTS 🌴
//lib injection code continued

//shellcode contains dlopen pointer callback
uint64_t addrOfDlopen = (uint64_t) dlopen;

//dylib is on disk
*path_to_dylib = "/path/to/mydylib"


//when remote thread executes

callBackFunction(*addrOfDlopen, *path_to_dylib)
```

Shellcode sets up a stack frame for a call to DLOPEN

Dylib Injection

```
➜  OBTS 🌴
//lib injection code continued

//shellcode contains dlopen pointer callback
uint64_t addrOfDlopen = (uint64_t) dlopen;

//dylib is on disk
*path_to_dylib = "/path/to/mydylib"


//when remote thread executes

callBackFunction(*addrOfDlopen, *path_to_dylib)
```

Target binary loads dylib in its own context, arbitrary code execution achieved

Subverting iCloud 2FA

- Codes are TOTP (i.e., solely determined by secret key material & wall-clock time)
- Hooks *gettimeofday* to "fool" *adid* about the current time
- Can generate 2FA codes valid for arbitrary future times!!!
- Plug & chug a ton of times into the injected *adid* ... profit!!



Complex Predicate Language

Complex Predicate Language

- VPN Connected (T/F)
- Proxy (T/F)
- Third-party Jailbreak (T/F)
- Device Attached (T/F)
- Battery Charging (T/F)
- Screen Locked (T/F)
- Battery Percentage (int)
- Battery Temp. Range (float)
- CPU Utilization (float)
- Located in Country (list)

Complex Predicate Language

- VPN Connected (T/F)
- Proxy (T/F)
- Third-party Jailbreak (T/F)
- Device Attached (T/F)
- Battery Charging (T/F)
- Screen Locked (T/F)
- Battery Percentage (int)
- Battery Temp. Range (float)
- CPU Utilization (float)
- Located in Country (list)
- Connectivity (Mobile Data/WiFi)
- Data Uploaded in Duration Exceeds Threshold
- Traveled to New Country
- Location within radius of coordinates
- Threatening process
- AND/OR/NOT/PIPE

Cleanup C&C Command...

- Step 1: Open Calendar.sqlitedb

Cleanup C&C Command...

- Step 1: Open Calendar.sqlitedb
- Step 2: Run queries, where %s is supplied by C&C:

```
DELETE FROM CalendarItemChanges WHERE record IN (SELECT
owner_id FROM ParticipantChanges WHERE email = "%s");

DELETE FROM ParticipantChanges WHERE email = "%s";

DELETE FROM Identity WHERE ROWID IN (SELECT DISTINCT
identity_id FROM Participant WHERE email = "%s");
```

Cleanup C&C Command...

- Step 1: Open Calendar.sqlitedb
- Step 2: Run queries, where %s is supplied by C&C:

```
DELETE FROM CalendarItemChanges WHERE record IN (SELECT
owner_id FROM ParticipantChanges WHERE email = "%s");

DELETE FROM ParticipantChanges WHERE email = "%s";

DELETE FROM Identity WHERE ROWID IN (SELECT DISTINCT
identity_id FROM Participant WHERE email = "%s");
```

- Step 3: Vacuum the DB



Discovery of the Attack & Samples

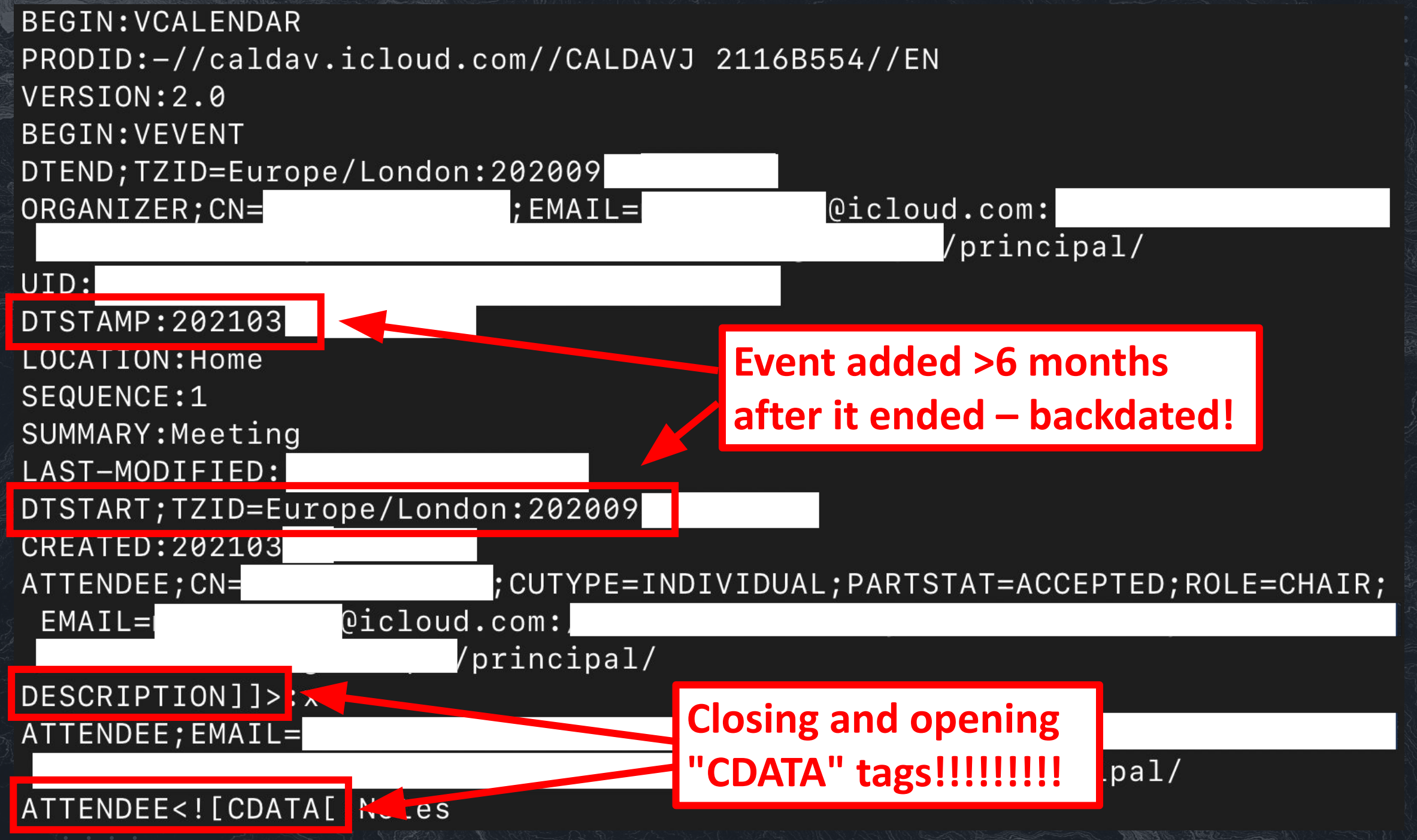
Attribution: Sometimes it's Easy!

Static & Dynamic Reversing of the Sample

A Theory of the Exploit

```
BEGIN:VCALENDAR
PRODID:-//caldav.icloud.com//CALDAVJ 2116B554//EN
VERSION:2.0
BEGIN:VEVENT
DTEND;TZID=Europe/London:202009
ORGANIZER;CN=;EMAIL=@icloud.com:
 /principal/
UID:
DTSTAMP:202103
LOCATION:Home
SEQUENCE:1
SUMMARY:Meeting
LAST-MODIFIED:
DTSTART;TZID=Europe/London:202009
CREATED:202103
ATTENDEE;CN=;CUTYPE=INDIVIDUAL;PARTSTAT=ACCEPTED;ROLE=CHAIR;
 EMAIL=@icloud.com:
 /principal/
DESCRIPTION]]>:x
ATTENDEE;EMAIL=;CN=:
 /principal/
ATTENDEE<![CDATA[:Notes
```


BEGIN:VCALENDAR
PRODID:-//caldav.icloud.com//CALDAVJ 2116B554//EN
VERSION:2.0
BEGIN:VEVENT
DTEND;TZID=Europe/London:202009
ORGANIZER;CN= ;EMAIL= @icloud.com:
/principal/
UID:
DTSTAMP:202103
LOCATION:Home
SEQUENCE:1
SUMMARY:Meeting
LAST-MODIFIED:
DTSTART;TZID=Europe/London:202009
CREATED:202103
ATTENDEE;CN= ;CUTYPE=INDIVIDUAL;PARTSTAT=ACCEPTED;ROLE=CHAIR;
EMAIL= @icloud.com:
/principal/
DESCRIPTION]]>:x
ATTENDEE;EMAIL= ;CN= :
/principal/
ATTENDEE<![CDATA[:Notes
Event added >6 months
after it ended – backdated!


BEGIN:VCALENDAR
PRODID:-//caldav.icloud.com//CALDAVJ 2116B554//EN
VERSION:2.0
BEGIN:VEVENT
DTEND;TZID=Europe/London:202009
ORGANIZER;CN=
;EMAIL=
@icloud.com:
/principal/
UID:
DTSTAMP:202103
LOCATION:Home
SEQUENCE:1
SUMMARY:Meeting
LAST-MODIFIED:
DTSTART;TZID=Europe/London:202009
CREATED:202103
ATTENDEE;CN=
;CUTYPE=INDIVIDUAL;PARTSTAT=ACCEPTED;ROLE=CHAIR;
EMAIL=
@icloud.com:
/principal/
DESCRIPTION]]>
ATTENDEE;EMAIL=
ATTENDEE<![CDATA[
Event added >6 months after it ended – backdated!
Closing and opening "CDATA" tags!!!!!!!!!

CDATA who???

```
<?xml version="1.0" encoding="utf-8"?>
    [...]
    <d:calendar-data><![CDATA[
        BEGIN:VCALENDAR
        [...]
        DESCRIPTION]]>:

        <lmao>parsed by the phone as XML</lmao>

        ATTENDEE<![CDATA[:Notes
    ]]></d:calendar-data>
    [...]
```

Hold up, does this really work?

Hold up, does this really work?

- Yes. Parsed by NSXMLParser (libxml2 SAX mode)

Hold up, does this really work?

- Yes. Parsed by NSXMLParser (libxml2 SAX mode)
- Hook the SAX callback when an element is found:

```
-[CoreDAVXMLElementGenerator
parser:didStartElement:namespaceURI:qualifiedName:attributes:]
```



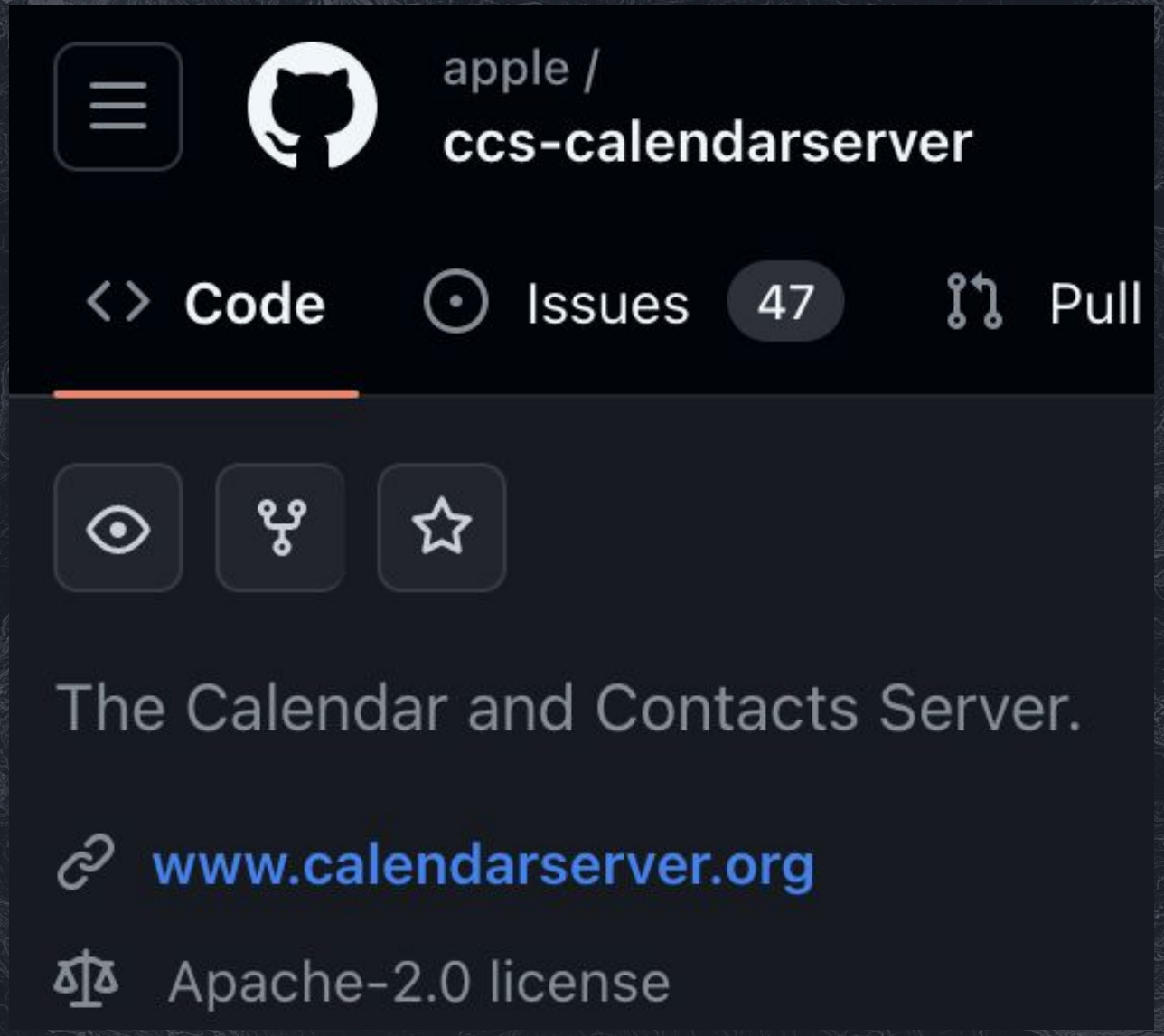
Phone's iCalendar Parser Only Saw CDATA!

```
<?xml version="1.0" encoding="utf-8"?>
    [...]
    <d:calendar-data><![CDATA[
        BEGIN:VCALENDAR
        [...]
        DESCRIPTION
                                    :Notes
    ]]></d:calendar-data>
    [...]
```

Can We Test Against a Server?



Can We Test Against a Server?



Can We Test Against a Server?

- Server rejects `]]>` and `<![CDATA[` in values (right of the ":") but accepts them in keys (left of the ":")



Can We Test Against a Server?

- Server rejects `]]>` and `<![CDATA[` in values (right of the ":") but accepts them in keys (left of the ":")



- Attacker can "update" to remove any XML escape
  - `DESCRIPTION]]>: <lmao>XML</lmao>`
  - `DESCRIPTION]]>: x`




BEGIN:VCALENDAR
PRODID:-//caldav.icloud.com//CALDAVJ 2116B554//EN
VERSION:2.0
BEGIN:VEVENT
DTEND;TZID=Europe/London:202009
ORGANIZER;CN=
;EMAIL=
@icloud.com:
/principal/
UID:
DTSTAMP:202103
LOCATION:Home
SEQUENCE:1
Oh yeah, updated once!
SUMMARY:Meeting
LAST-MODIFIED:
DTSTART;TZID=Europe/London:202009
CREATED:202103
ATTENDEE;CN=
;CUTYPE=INDIVIDUAL;PARTSTAT=ACCEPTED;ROLE=CHAIR;
EMAIL=
@icloud.com:
/principal/
DESCRIPTION]]>:x
ATTENDEE;EMAIL=
;CN=
:
/principal/
ATTENDEE<![CDATA[:Notes




AIR

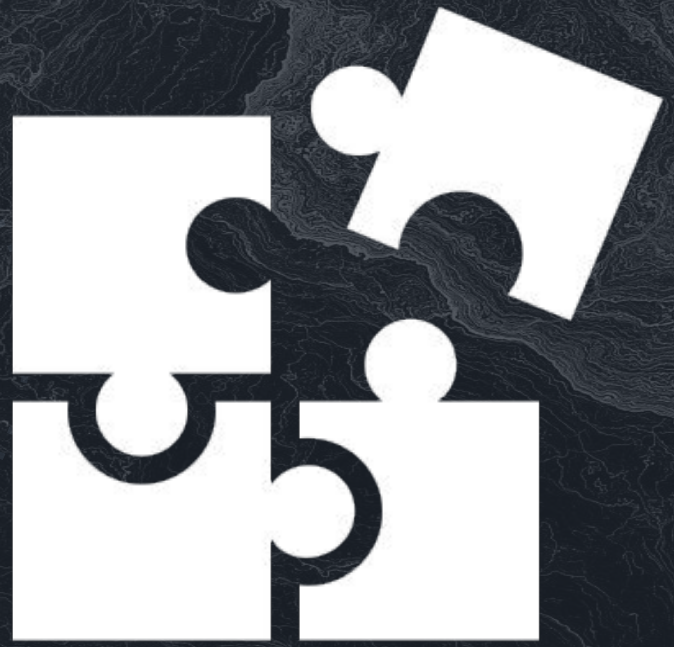



JUL
17

Conclusions

Conclusions

Collaboration and information sharing is important: include civil society too!



Conclusions

Cloud services as new vector, beyond the classics.



Conclusions

Build the wall broader, not just taller in one place.



Conclusions

Features like Lockdown Mode are great, but optional.



Questions?

bill@citizenlab.ca
cfossaceca@microsoft.com

A Special Thank You To



Objective by the Sea

THECITIZENLAB

Microsoft