SiDi

Linux to Android: a path to becoming a secure mobile operating system

Danilo Rodrigues

$ whoami

SiDi

Cloud

Machine learning

Security

What will be discussed today?

- What exactly is Android?
- Is Android secure?
  - Linux inherited security features
  - Android's own security features
- What are the security issues on Android today?
- BONUS: How Brazilian authorities were hacked?

Meet Android

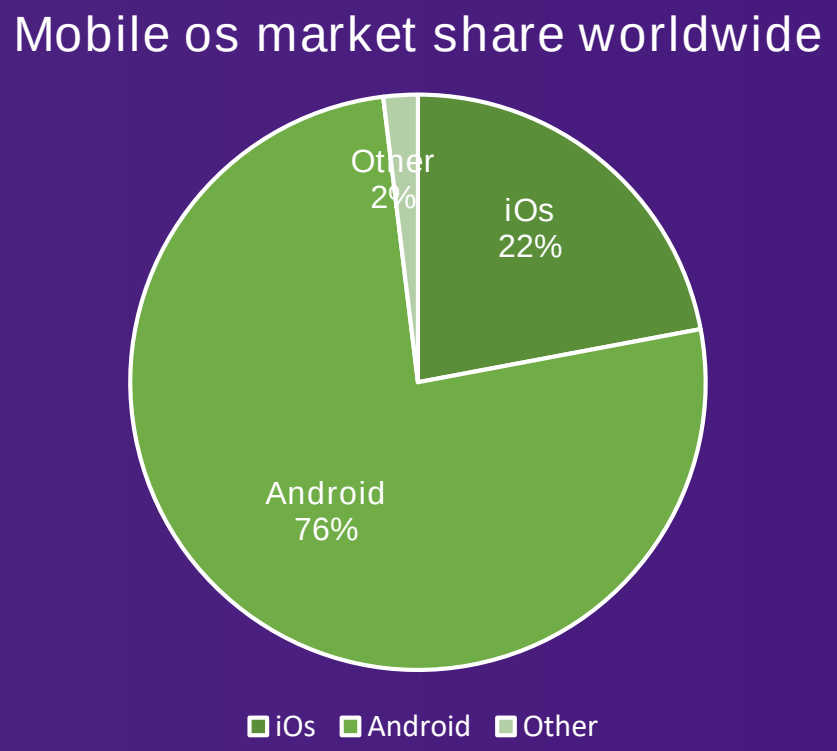
In the beginning...

The FIRST Android_

T-Mobile G1, the first Android phone sold

ANDROID today.

2.5 BILLION DEVICES
2.1 MILLION APPS
5.9 MILLION DEVELOPERS



Linux and Android

Is Android another Linux distribution?

Maybe a little bit Linux

Common Features

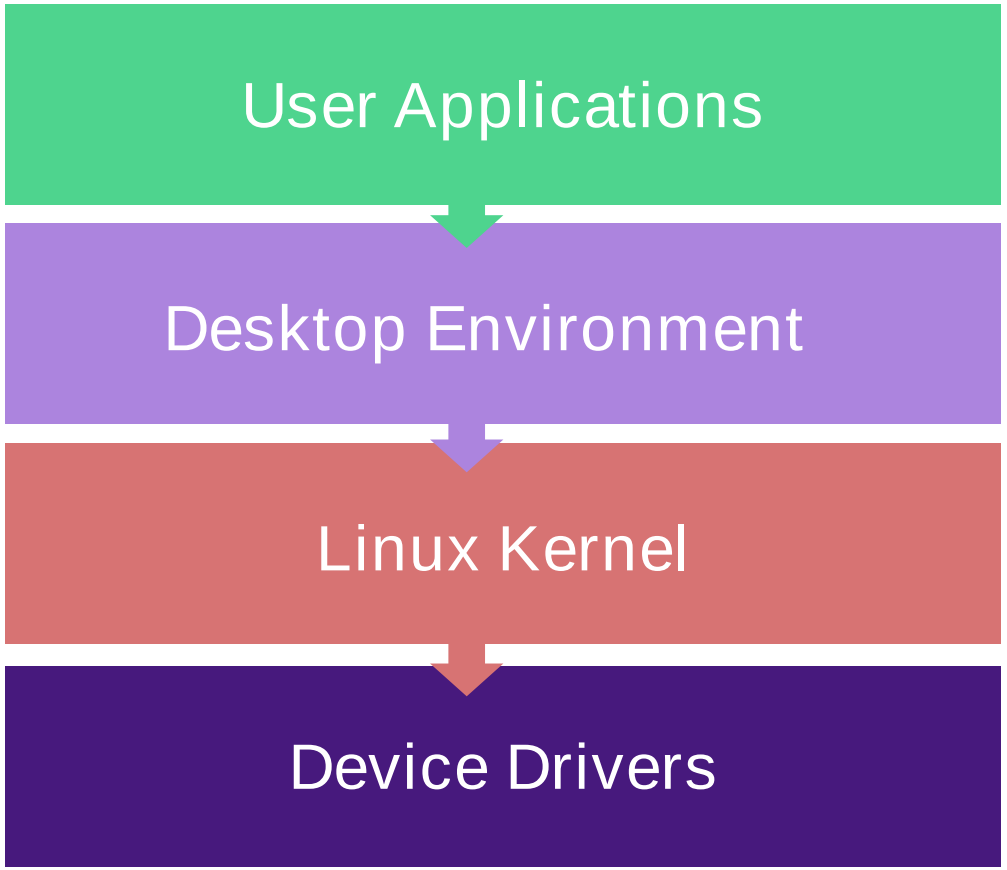
- Many native binaries
- Same process and thread behavior
- Control groups
- Low memory conditions
- Security features

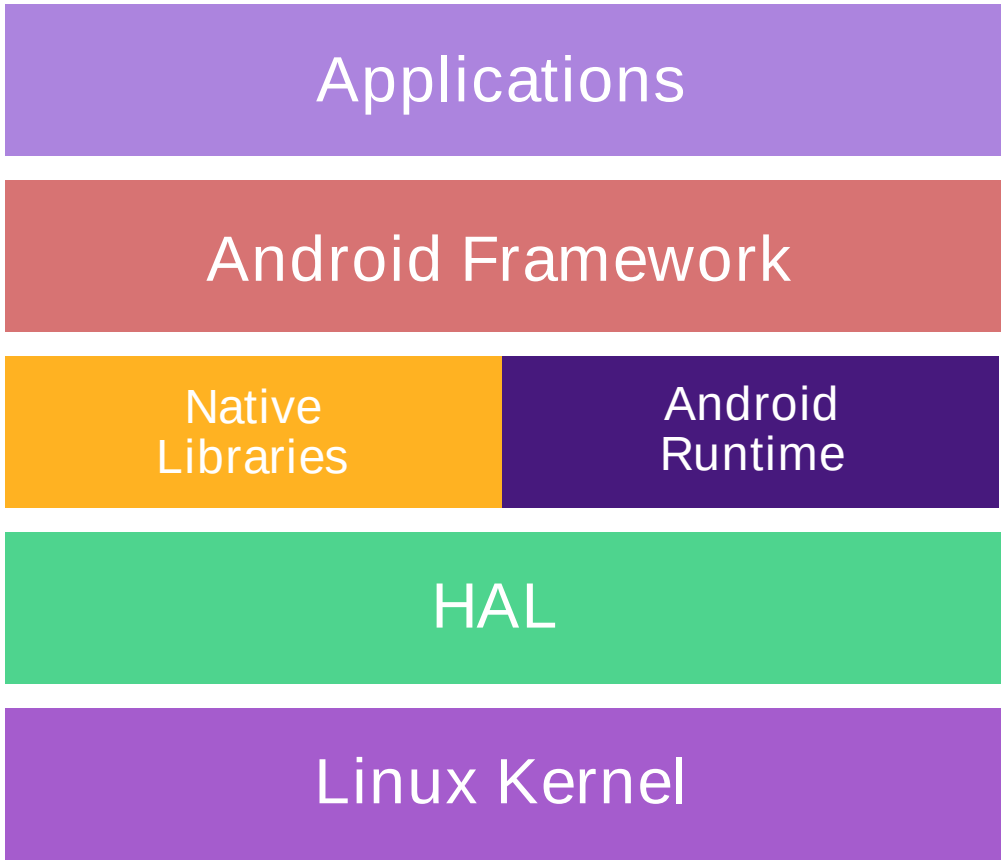
Android and Linux are about 95% equal in Kernel level!

Different problems require different solutions.

Desktop

Mobile



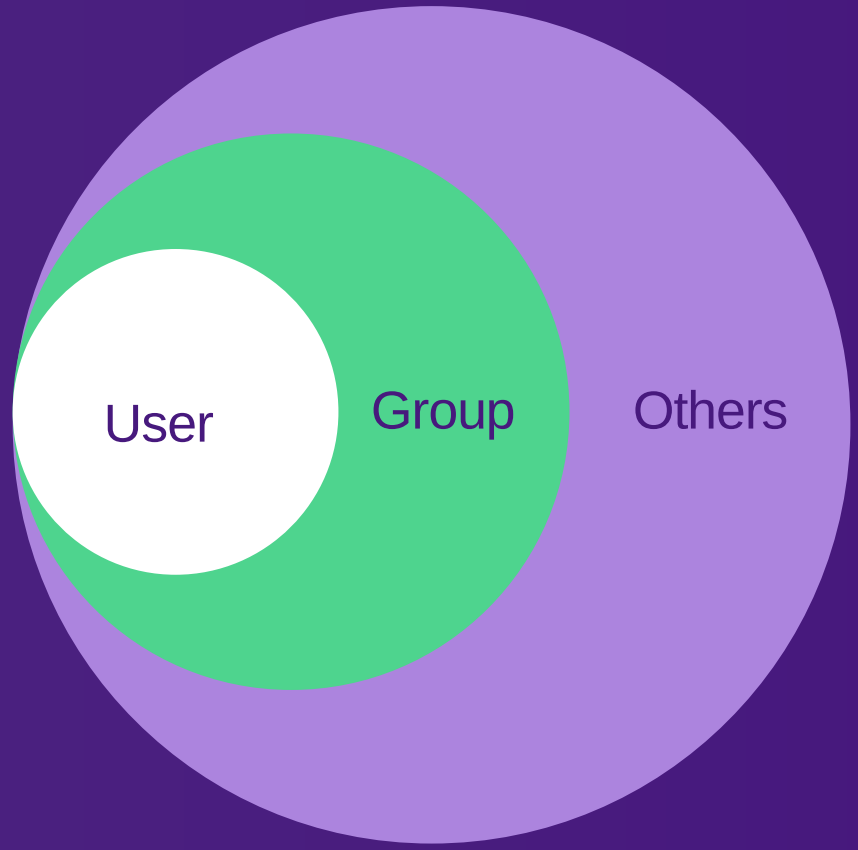


Using Linux *security* features.

Linux permissions

-rwxr-xr-x

r: Read
w: Write
x: eXecute



Application A

UID: 10209 (u0_a209)
GUID: 10209 (u0_a209)

/data/data/applicationA

Application B

UID: 10210 (u0_a210)
GUID: 10210 (u0_a210)

/data/data/applicationB

Application A

UID: 10209 (u0_a209)
GUID: 10209 (u0_a209)

/data/data/applicationA

Application B

UID: 10210 (u0_a210)
GUID: 10210 (u0_a210)

/data/data/applicationB

Application A

UID: 10209 (u0_a209)
GUID: 10209 (u0_a209)

/data/data/applicationA

Application B

UID: 10209 (u0_a209)
GUID: 10209 (u0_a209)

/data/data/applicationB

Linux permissions

“All apps are equal, but some apps are more equal than others.”

Process	UID
ROOT	0
…	…
SYSTEM SERVER	1000
RADIO	1001
BLUETOOTH	1002
GRAPHICS	1003
…	…
SHELL	2000
…	…
Regular APP 0	10000
Regular APP 1	10001
…	…

Linux capabilities

You DONT have to be root to do root stuff!

```
[mgn@lois ~]$ getcap /bin/ping                                  # Getting ping capabilities
/bin/ping = cap_net_raw+ep
[mgn@lois ~]$ sudo setcap -r /bin/ping                          # Removing ping capabilities
[mgn@lois ~]$ ping google.com                                   # Testing ping...
ping: socket: Operação não permitida
[mgn@lois ~]$ sudo ping google.com                              # Now testing as root...
PING google.com (172.217.28.78) 56(84) bytes of data.
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=1 ttl=251 time=11.7 ms
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=2 ttl=251 time=12.6 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 11.744/12.172/12.601/0.442 ms
[mgn@lois ~]$ sudo setcap 'cap_net_raw+ep' /bin/ping            # Restoring ping capability
[mgn@lois ~]$ ping google.com                                   # Retesting ping...
PING google.com (172.217.28.78) 56(84) bytes of data.
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=1 ttl=251 time=23.1 ms
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=2 ttl=251 time=10.3 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1ms
rtt min/avg/max/mdev = 10.315/16.705/23.095/6.390 ms
[mgn@lois ~]$
```

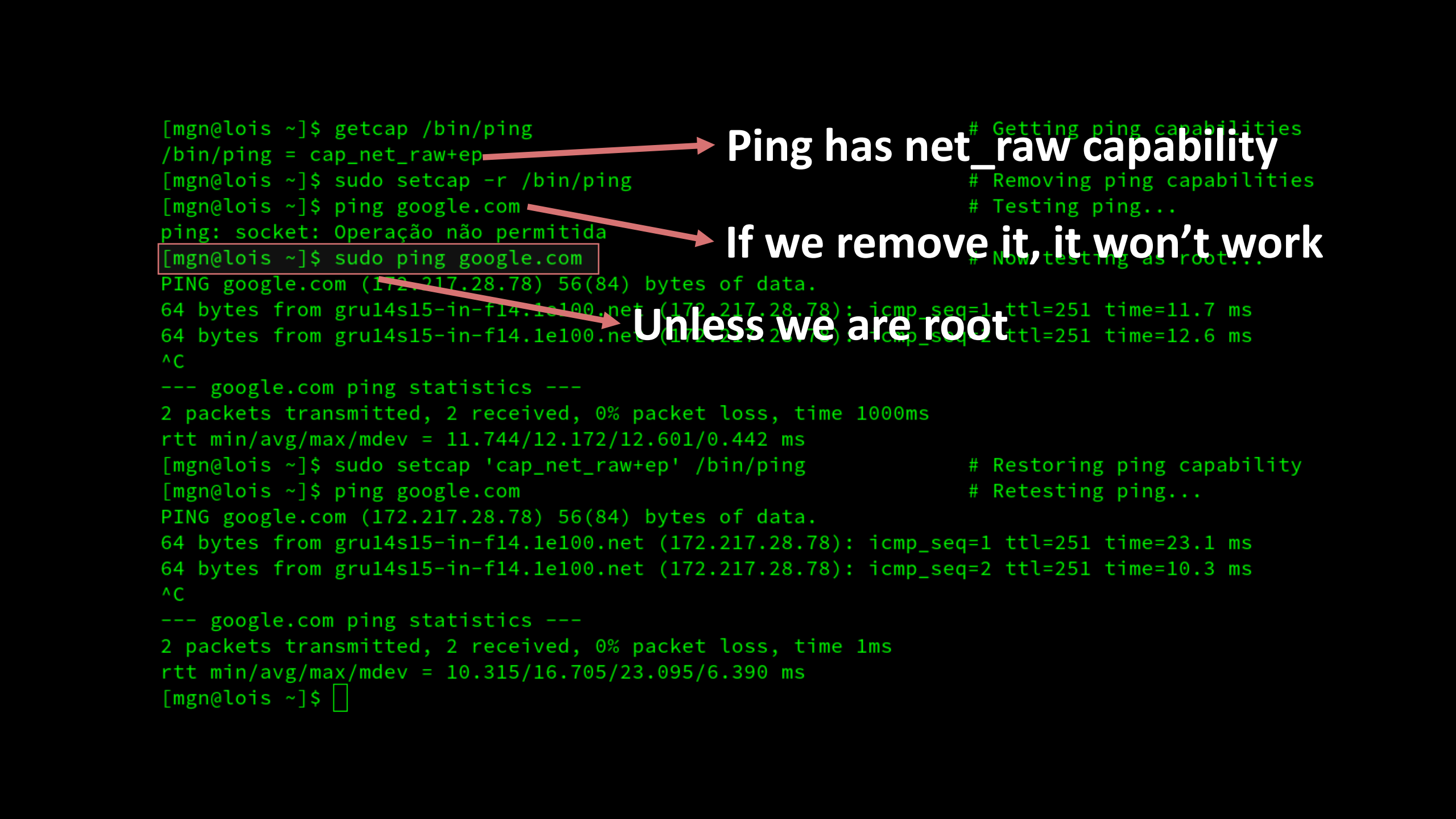
```
[mgn@lois ~]$ getcap /bin/ping                          # Getting ping capabilities
/bin/ping = cap_net_raw+ep
[mgn@lois ~]$ sudo setcap -r /bin/ping                  # Removing ping capabilities
[mgn@lois ~]$ ping google.com                           # Testing ping...
ping: socket: Operação não permitida
[mgn@lois ~]$ sudo ping google.com                      # Now testing as root...
PING google.com (172.217.28.78) 56(84) bytes of data.
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=1 ttl=251 time=11.7 ms
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=2 ttl=251 time=12.6 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 11.744/12.172/12.601/0.442 ms
[mgn@lois ~]$ sudo setcap 'cap_net_raw+ep' /bin/ping    # Restoring ping capability
[mgn@lois ~]$ ping google.com                           # Retesting ping...
PING google.com (172.217.28.78) 56(84) bytes of data.
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=1 ttl=251 time=23.1 ms
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=2 ttl=251 time=10.3 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1ms
rtt min/avg/max/mdev = 10.315/16.705/23.095/6.390 ms
[mgn@lois ~]$
```

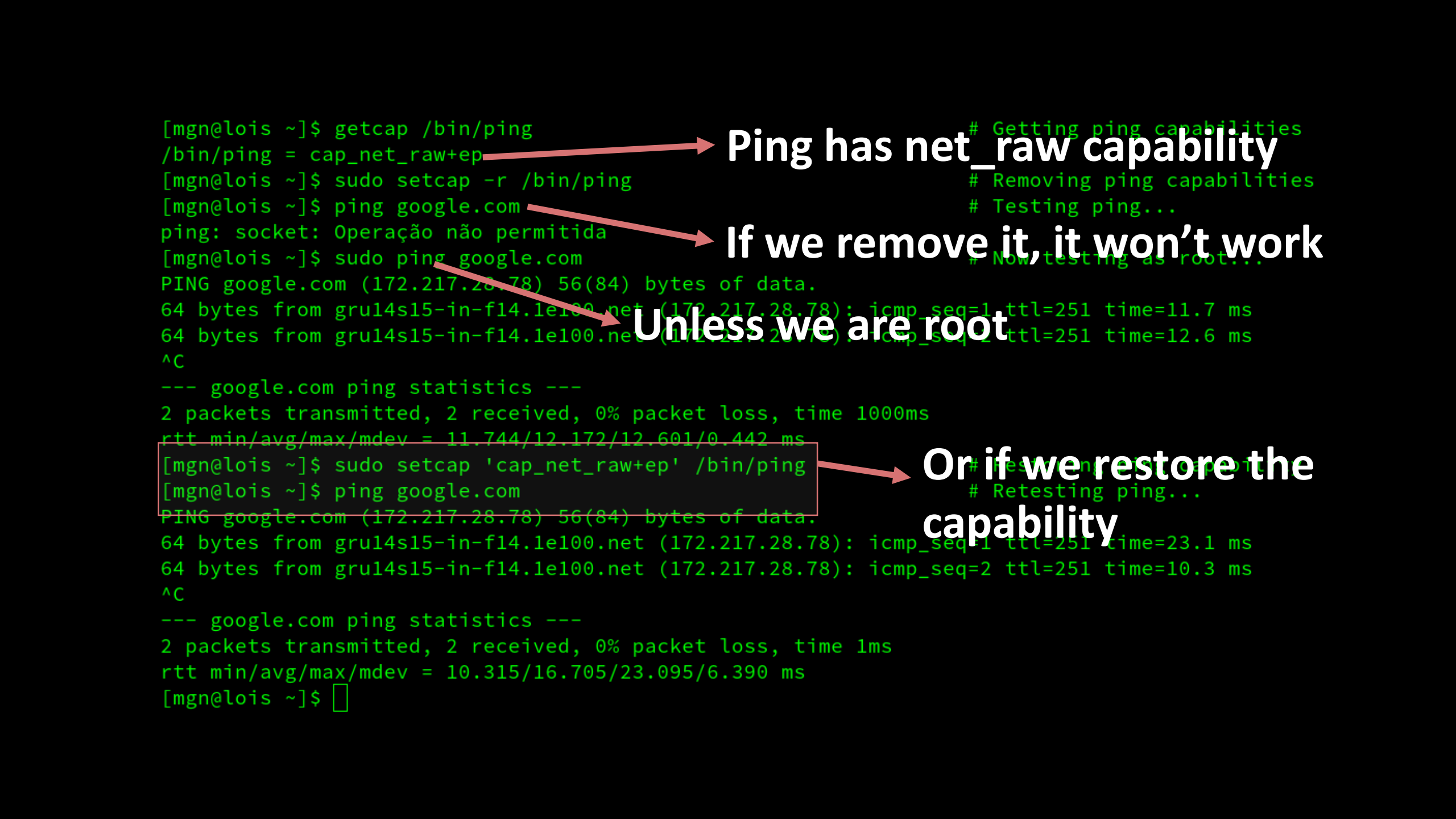
Ping has net_raw capability

```
[mgn@lois ~]$ getcap /bin/ping                          # Getting ping capabilities
/bin/ping = cap_net_raw+ep
[mgn@lois ~]$ sudo setcap -r /bin/ping                  # Removing ping capabilities
[mgn@lois ~]$ ping google.com                           # Testing ping...
ping: socket: Operação não permitida
[mgn@lois ~]$ sudo ping google.com                      # Now testing as root...
PING google.com (172.217.28.78) 56(84) bytes of data.
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=1 ttl=251 time=11.7 ms
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=2 ttl=251 time=12.6 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 11.744/12.172/12.601/0.442 ms
[mgn@lois ~]$ sudo setcap 'cap_net_raw+ep' /bin/ping    # Restoring ping capability
[mgn@lois ~]$ ping google.com                           # Retesting ping...
PING google.com (172.217.28.78) 56(84) bytes of data.
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=1 ttl=251 time=23.1 ms
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=2 ttl=251 time=10.3 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1ms
rtt min/avg/max/mdev = 10.315/16.705/23.095/6.390 ms
[mgn@lois ~]$
```

Ping has net_raw capability

If we remove it, it won't work


[mgn@lois ~]$ getcap /bin/ping
/bin/ping = cap_net_raw+ep
[mgn@lois ~]$ sudo setcap -r /bin/ping
[mgn@lois ~]$ ping google.com
ping: socket: Operação não permitida
[mgn@lois ~]$ sudo ping google.com
PING google.com (172.217.28.78) 56(84) bytes of data.
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=1 ttl=251 time=11.7 ms
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=2 ttl=251 time=12.6 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 11.744/12.172/12.601/0.442 ms
[mgn@lois ~]$ sudo setcap 'cap_net_raw+ep' /bin/ping
[mgn@lois ~]$ ping google.com
PING google.com (172.217.28.78) 56(84) bytes of data.
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=1 ttl=251 time=23.1 ms
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=2 ttl=251 time=10.3 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1ms
rtt min/avg/max/mdev = 10.315/16.705/23.095/6.390 ms
[mgn@lois ~]$
Getting ping capabilities
Removing ping capabilities
Testing ping...
Now testing as root...
Restoring ping capability
Retesting ping...
Ping has net_raw capability
If we remove it, it won't work
Unless we are root


[mgn@lois ~]$ getcap /bin/ping
/bin/ping = cap_net_raw+ep
[mgn@lois ~]$ sudo setcap -r /bin/ping
[mgn@lois ~]$ ping google.com
ping: socket: Operação não permitida
[mgn@lois ~]$ sudo ping google.com
PING google.com (172.217.28.78) 56(84) bytes of data.
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=1 ttl=251 time=11.7 ms
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=2 ttl=251 time=12.6 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
rtt min/avg/max/mdev = 11.744/12.172/12.601/0.442 ms
[mgn@lois ~]$ sudo setcap 'cap_net_raw+ep' /bin/ping
[mgn@lois ~]$ ping google.com
PING google.com (172.217.28.78) 56(84) bytes of data.
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=1 ttl=251 time=23.1 ms
64 bytes from gru14s15-in-f14.1e100.net (172.217.28.78): icmp_seq=2 ttl=251 time=10.3 ms
^C
--- google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1ms
rtt min/avg/max/mdev = 10.315/16.705/23.095/6.390 ms
[mgn@lois ~]$
Getting ping capabilities
Removing ping capabilities
Testing ping...
Now testing as root...
Retesting ping...
Ping has net_raw capability
If we remove it, it won't work
Unless we are root
Or if we restore the capability

Linux capabilities_

Android's processes drop privileges before they do anything.

- Principle of least privilege
- Sandboxed capabilities
- Lower security risk

SELinux_

Discretionary Access Control

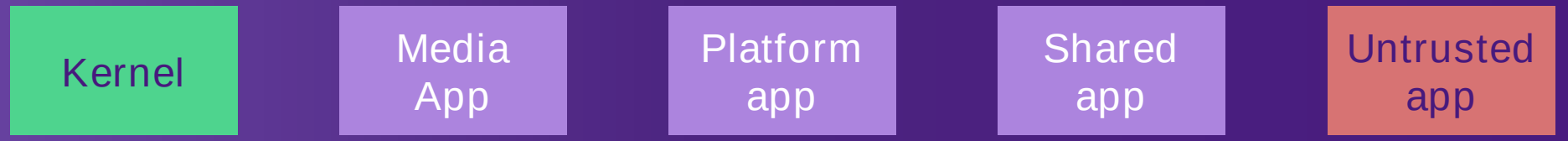
VS

Mandatory Access Control

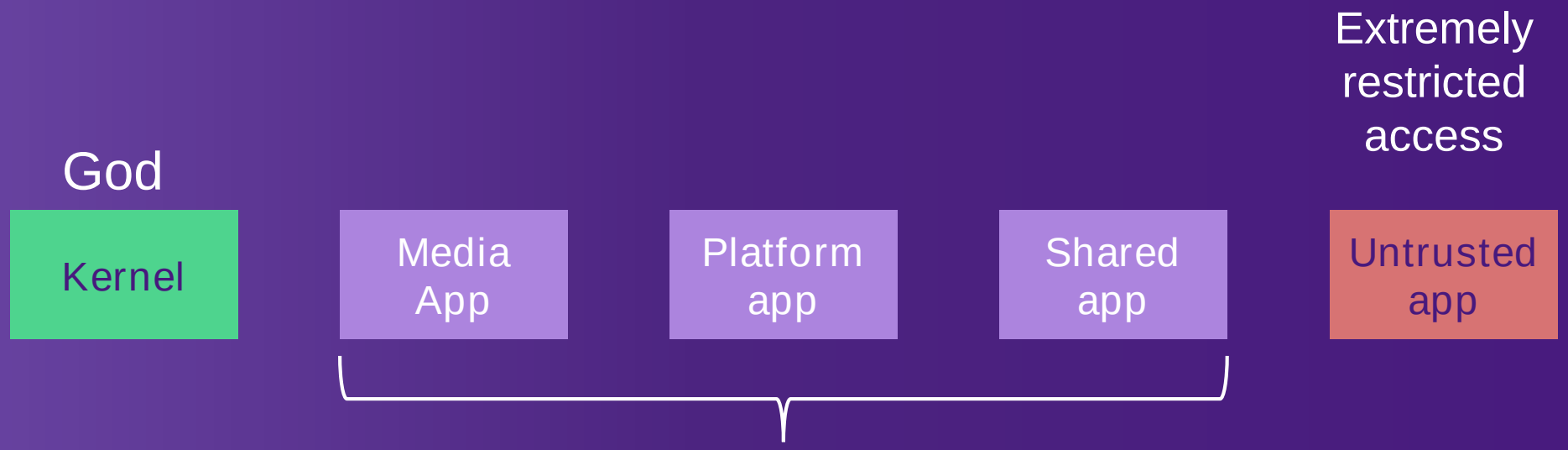
USER:ROLE:TYPE:LEVEL

~~USER~~:~~ROLE~~:TYPE:~~LEVEL~~

~~USER~~:~~ROLE~~:TYPE:~~LEVEL~~



~~USER~~:~~ROLE~~:TYPE:~~LEVEL~~



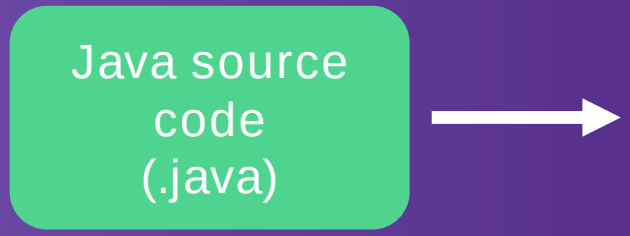
Other Linux protections_

- Address Space Layout Randomization
- Kernel hardening
- Stack protections
- Data execution prevention

Android Runtime security features_

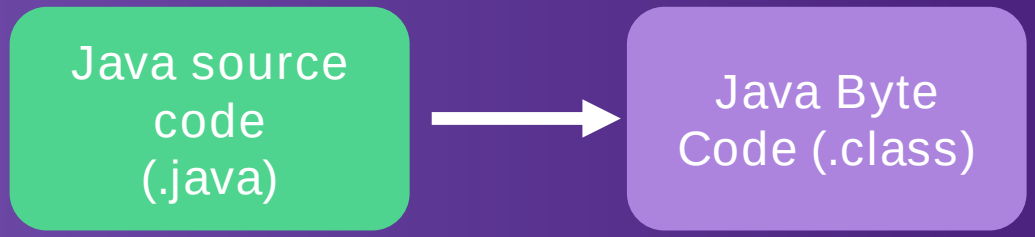
Running Java

How it’s done in Java



Running Java

How it's done in Java



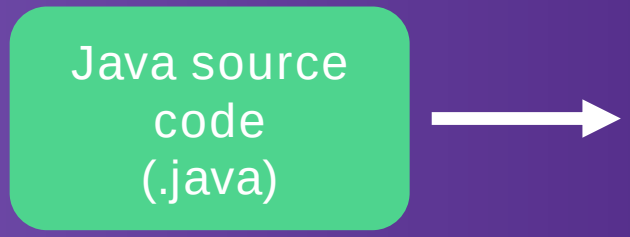
Running Java

How it's done in Java

Running Java

How it's done in Java

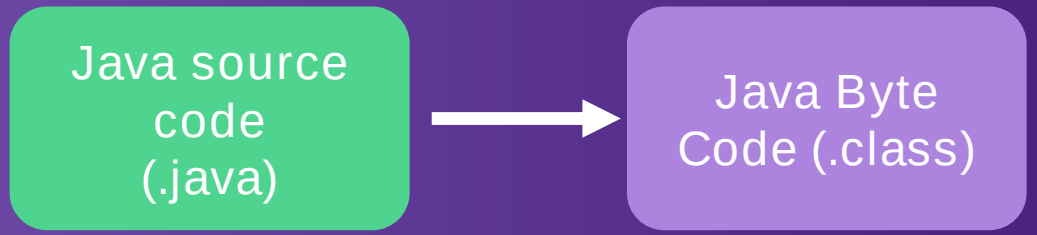
How Android did it



Running Java

How it's done in Java

How Android did it



Running Java

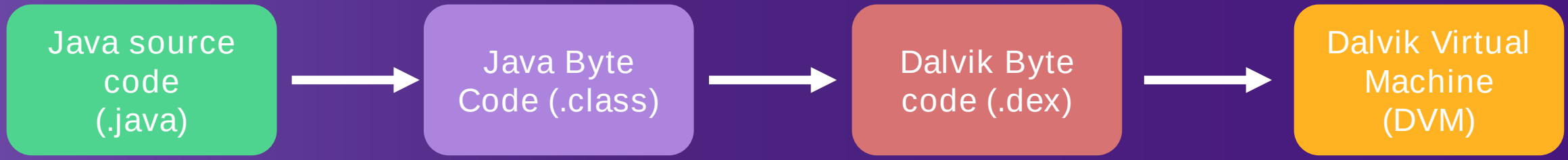
How it's done in Java

How Android did it

Running Java

How it's done in Java

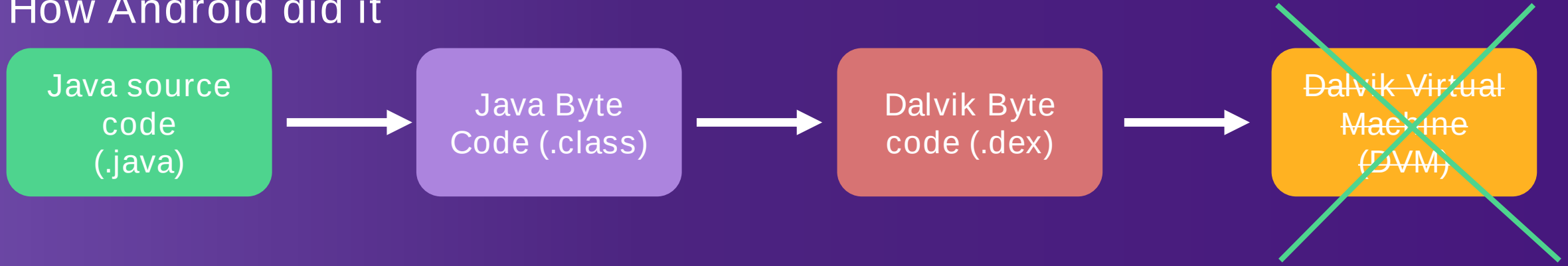
How Android did it



Running Java

How it's done in Java

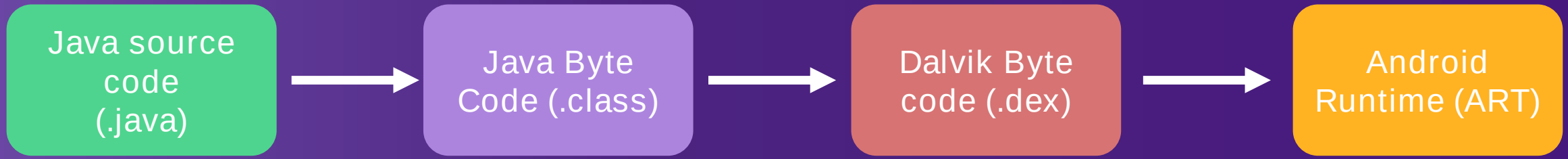
How Android did it



Running Java

How it's done in Java

How Android did it



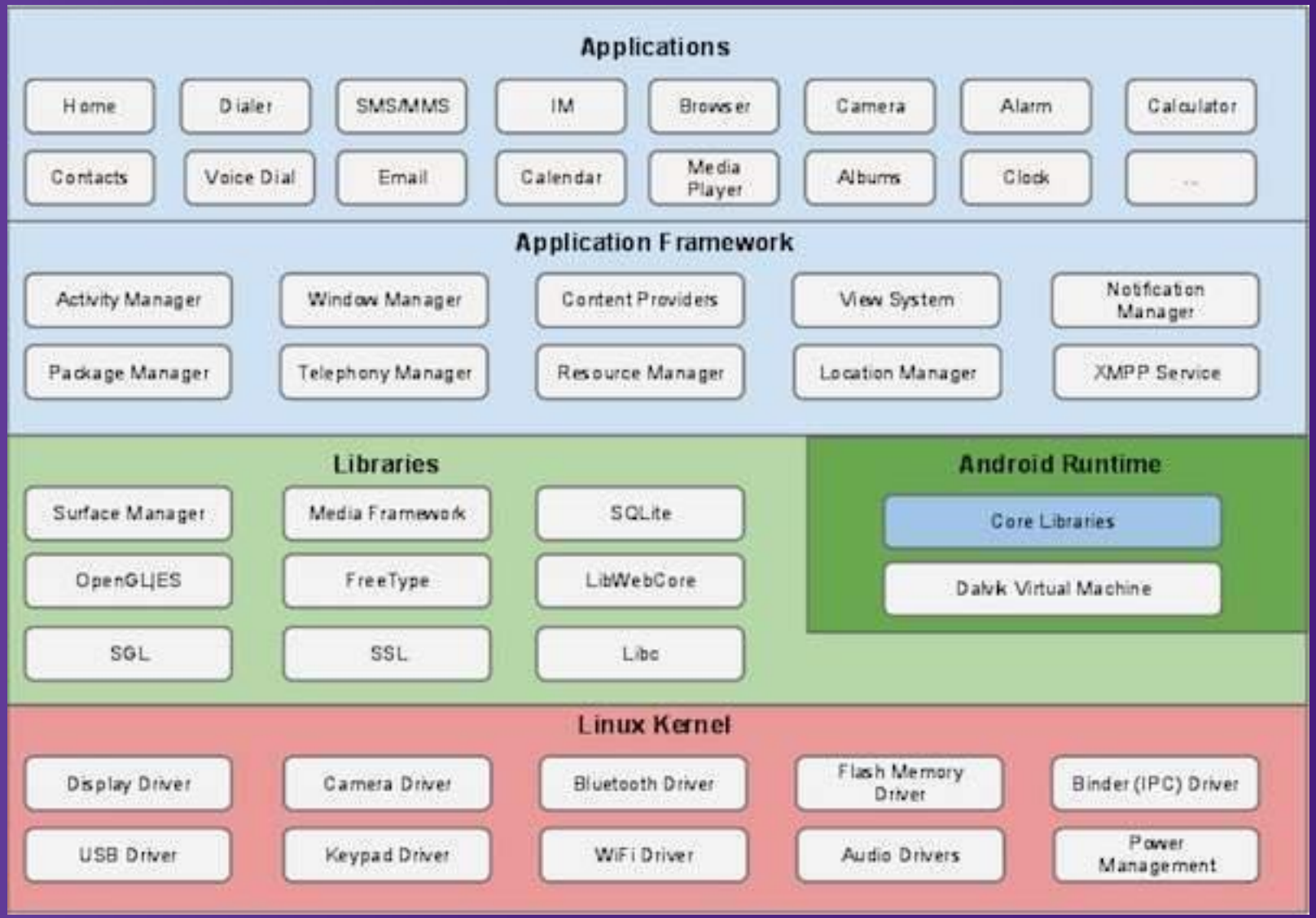
Android Runtime_

- Virtualization
- Granular permissions (AndroidManifest.xml)
- Code signing
- App has no direct system access

Android Runtime_

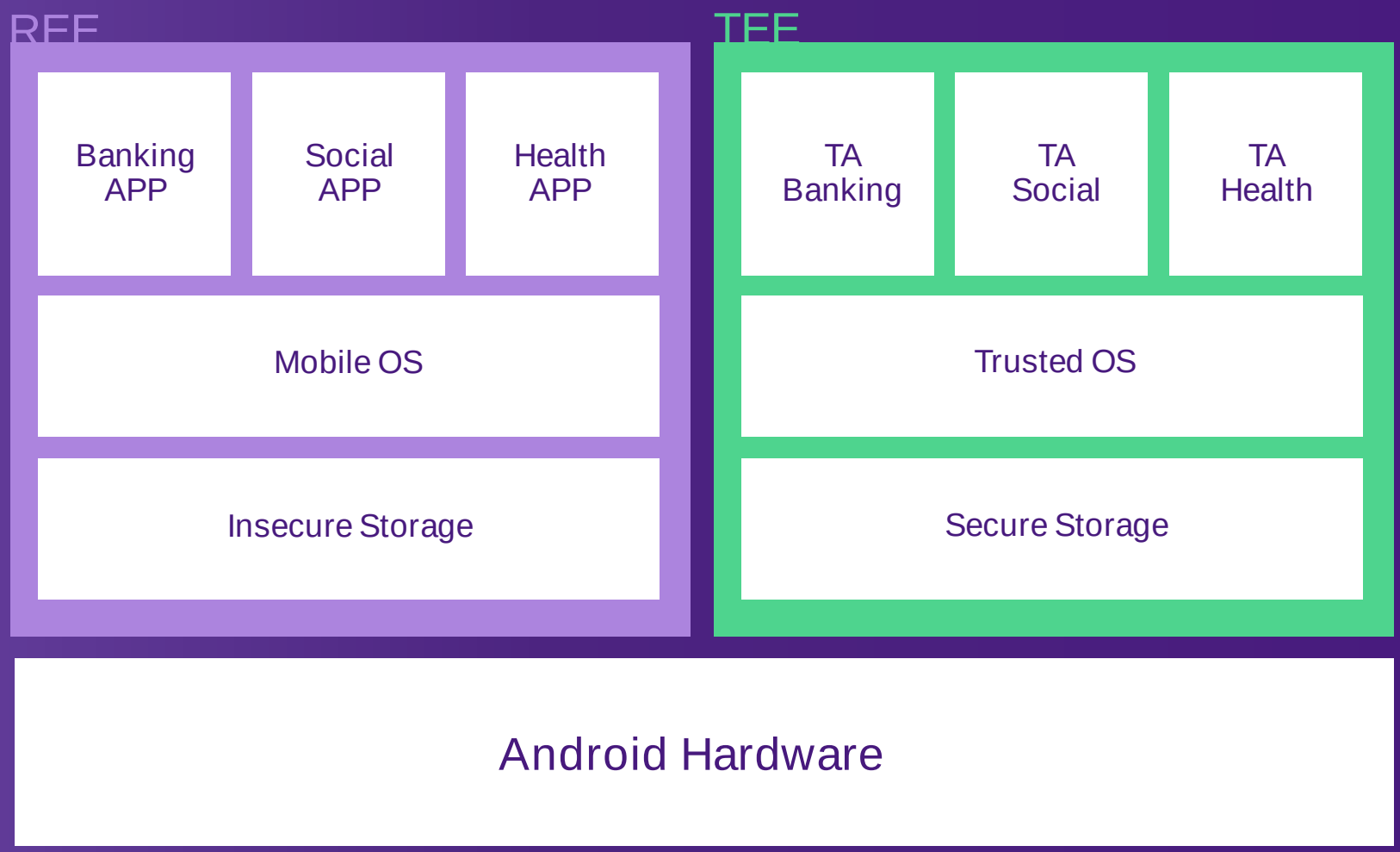
- Virtualization
- Granular permissions (AndroidManifest.xml)
- Code signing
- App has no direct system access

Android Framework

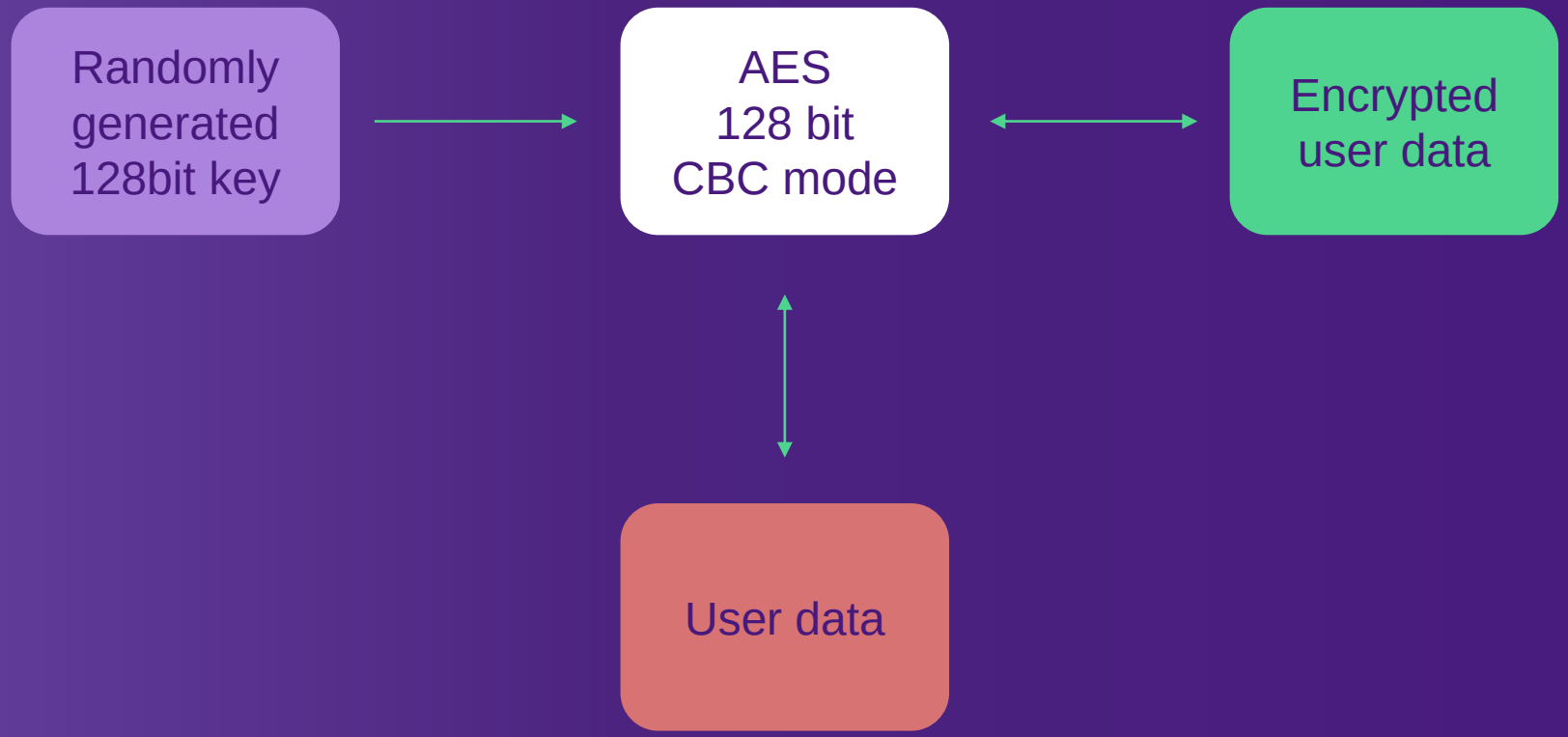


Other SECURITY features

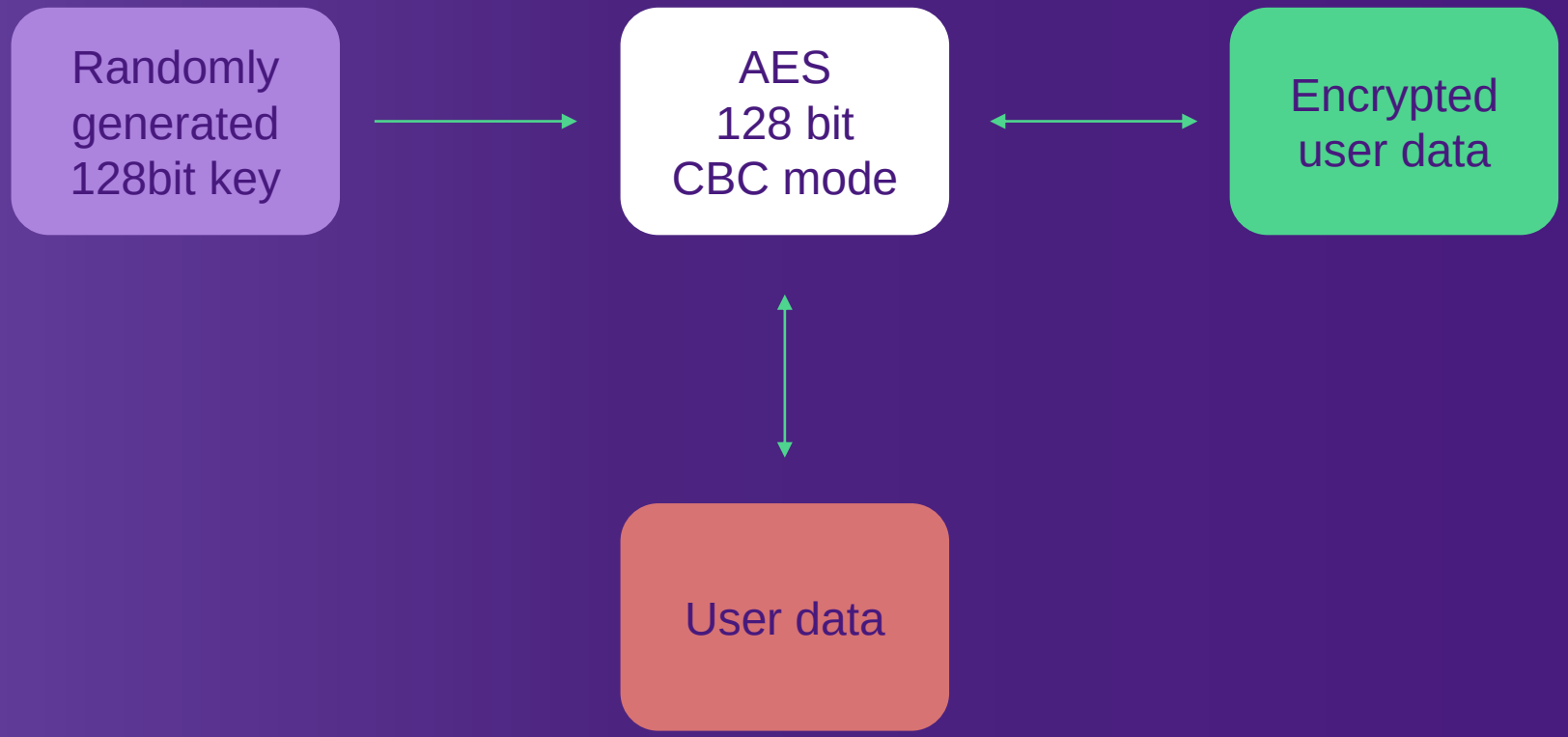
Trusted Execution Environment.



Full disk encryption_

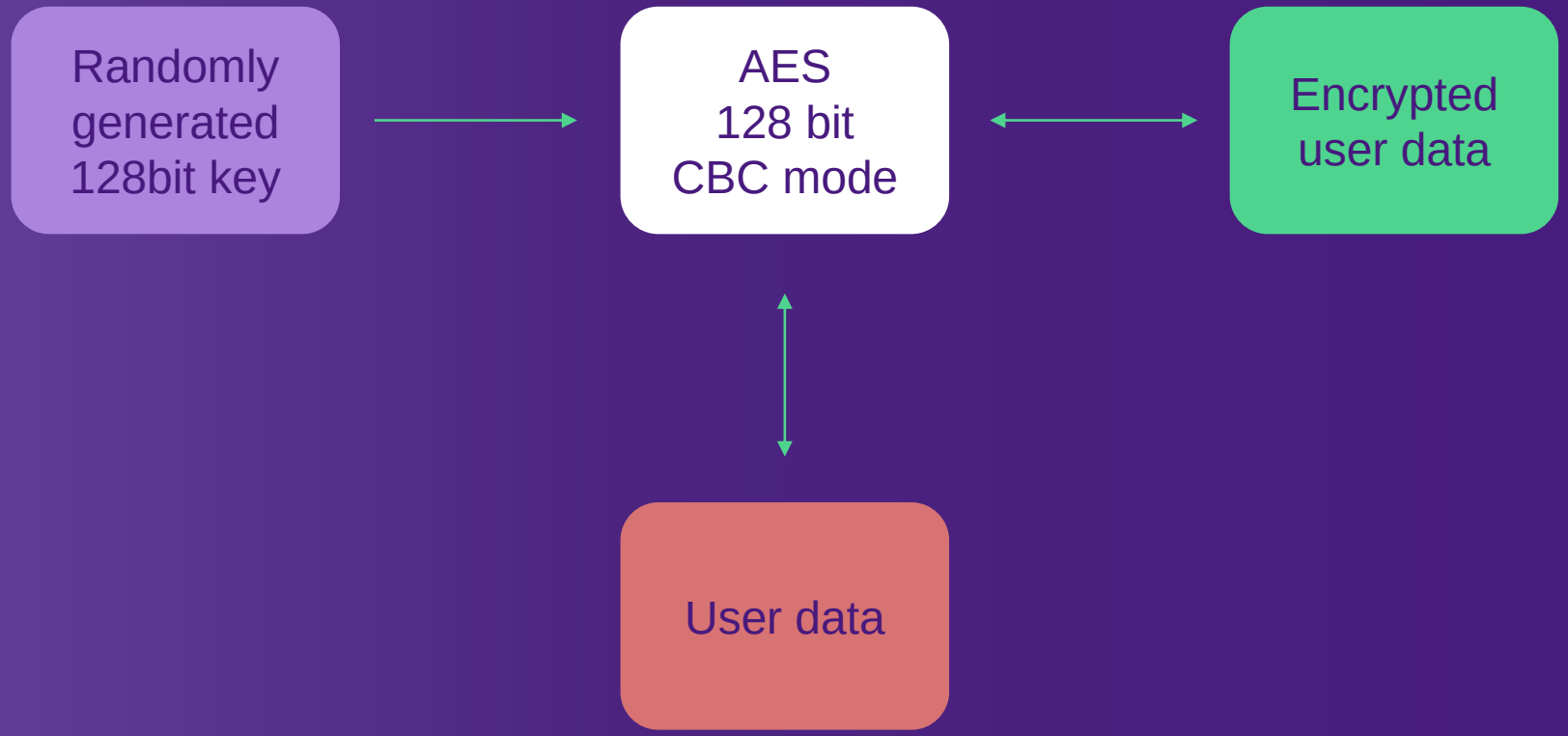


How do I securely store this key?



How do I securely store this key?

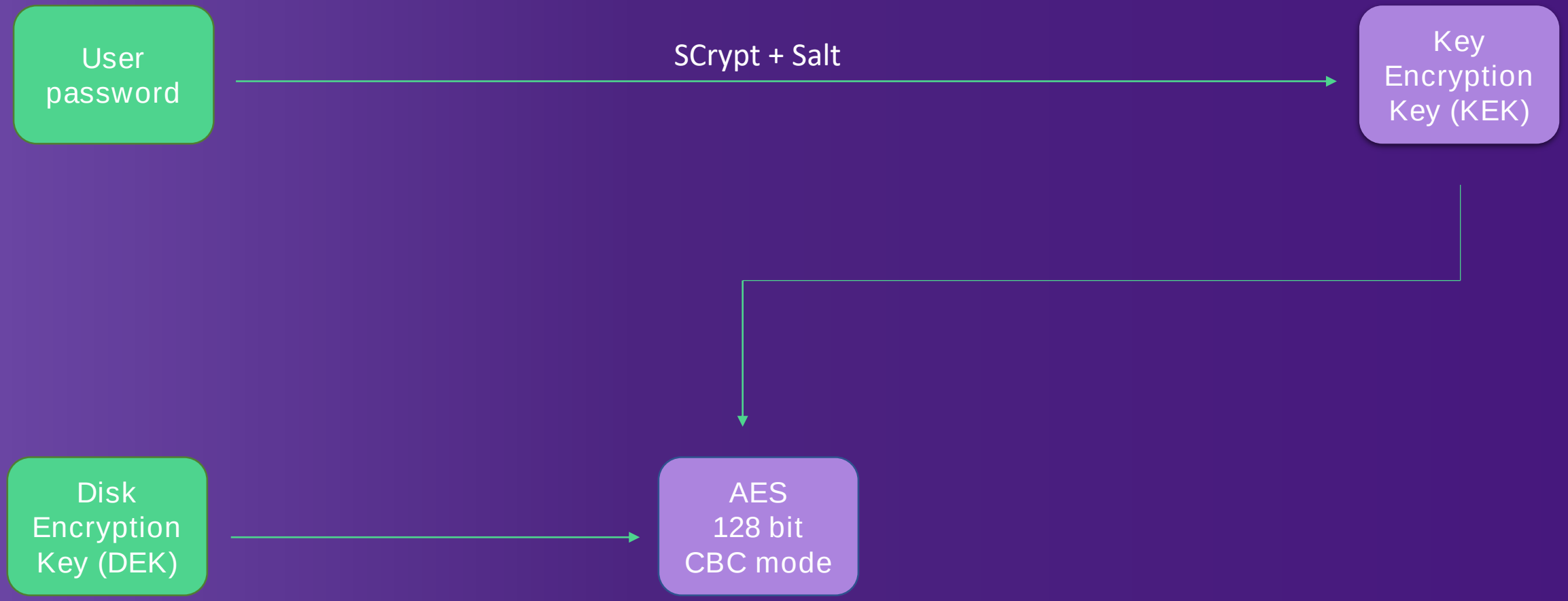
What about user pin/password?

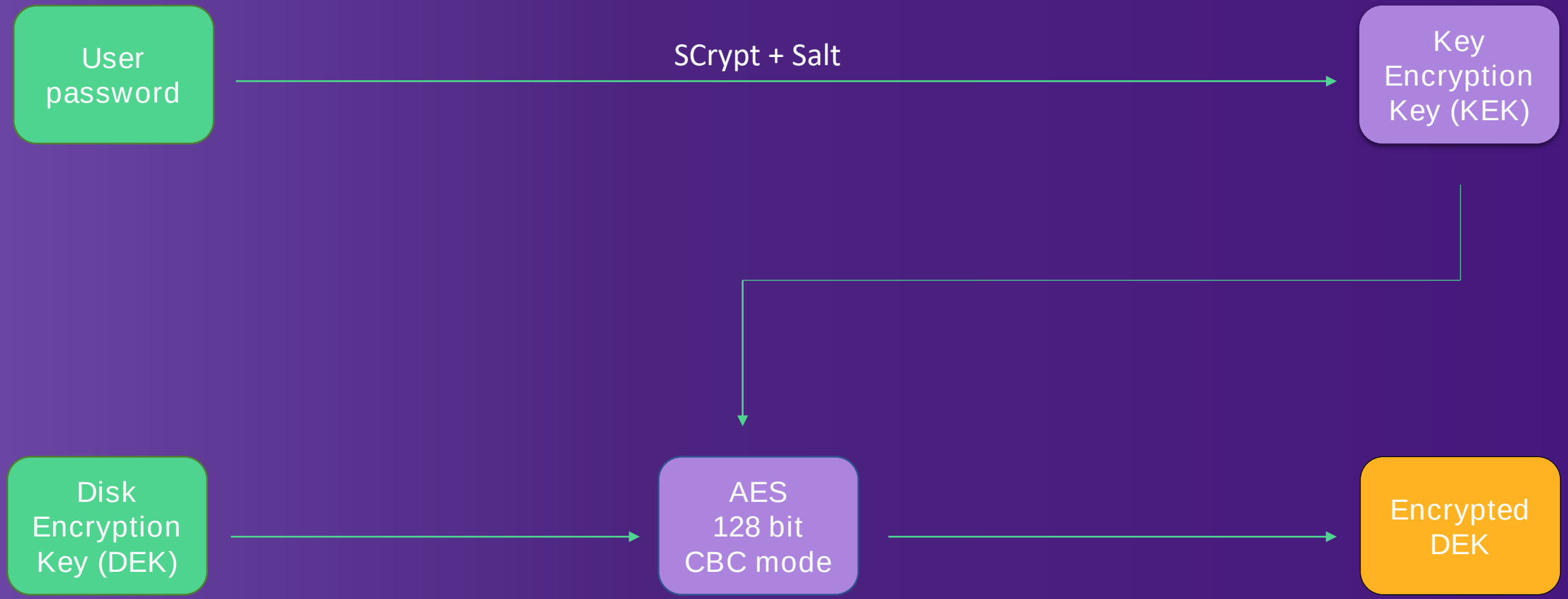


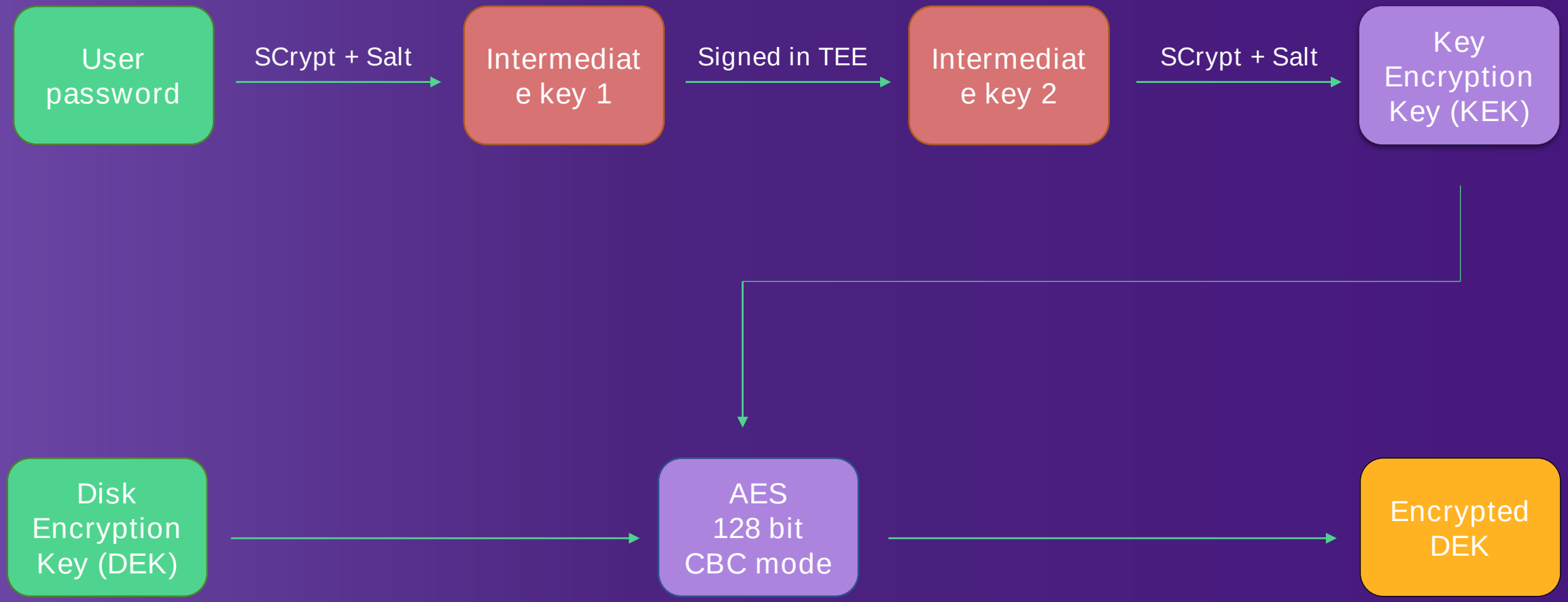
Full disk encryption_

Randomly generated 128bit key

Disk Encryption Key (DEK)


User password
SCrypt + Salt
Key Encryption Key (KEK)
Disk Encryption Key (DEK)
AES 128 bit CBC mode

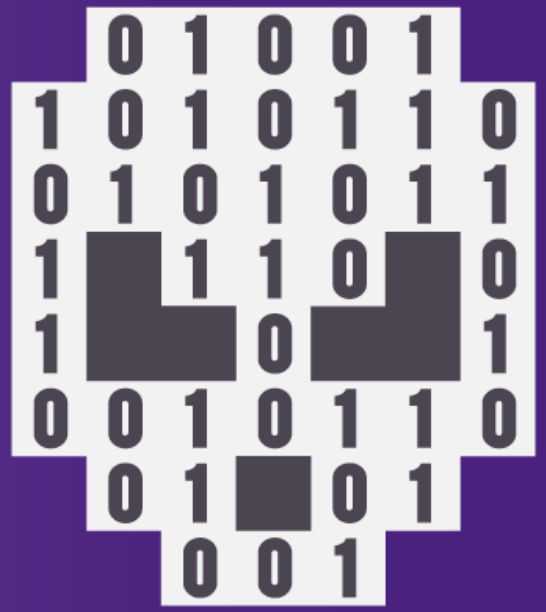

User password
SCrypt + Salt
Key Encryption Key (KEK)
Disk Encryption Key (DEK)
AES 128 bit CBC mode
Encrypted DEK


User password
SCrypt + Salt
Intermediate key 1
Signed in TEE
Intermediate key 2
SCrypt + Salt
Key Encryption Key (KEK)
Disk Encryption Key (DEK)
AES 128 bit CBC mode
Encrypted DEK

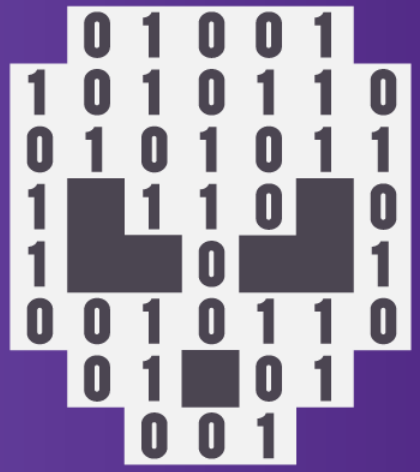
KEYSTORE

Attack surfaces_

Malicious App



Malicious App

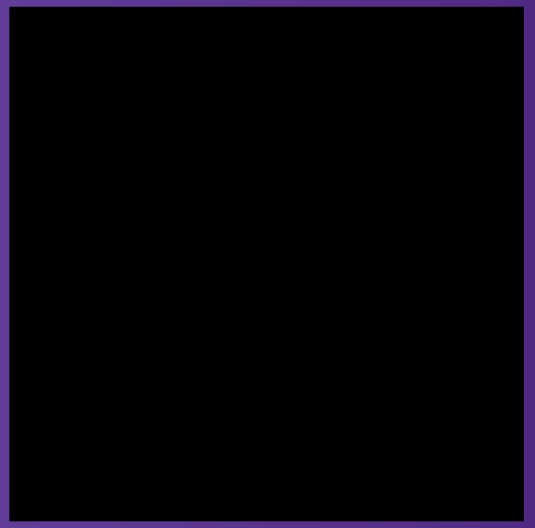


- **Android Runtime**

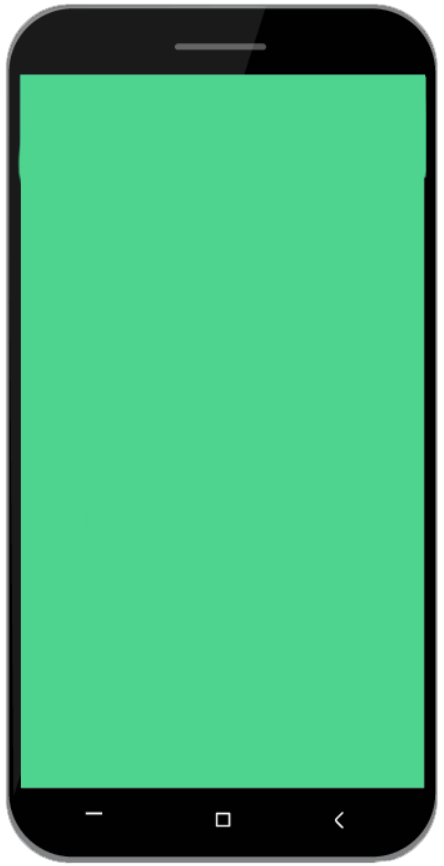
Malicious App

- **Android Runtime**
- **Linux permissions**

Malicious App



- **Android Runtime**
- **Linux permissions**
- **SELinux**



Physical attack_

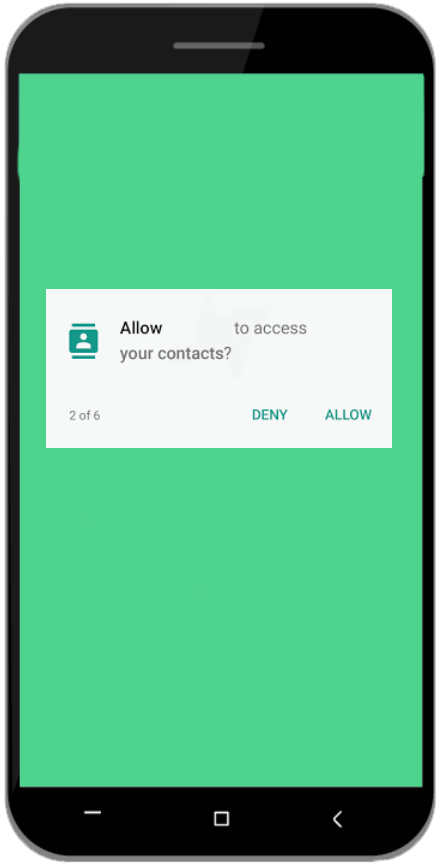
Physical attack_

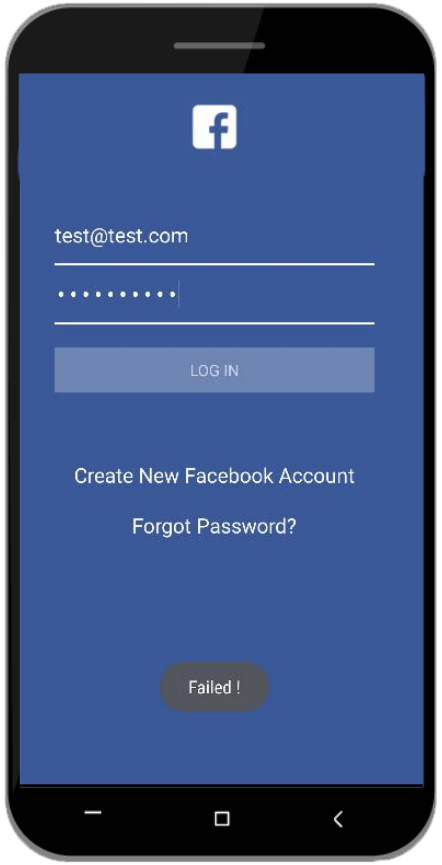
- **Full disk encryption**

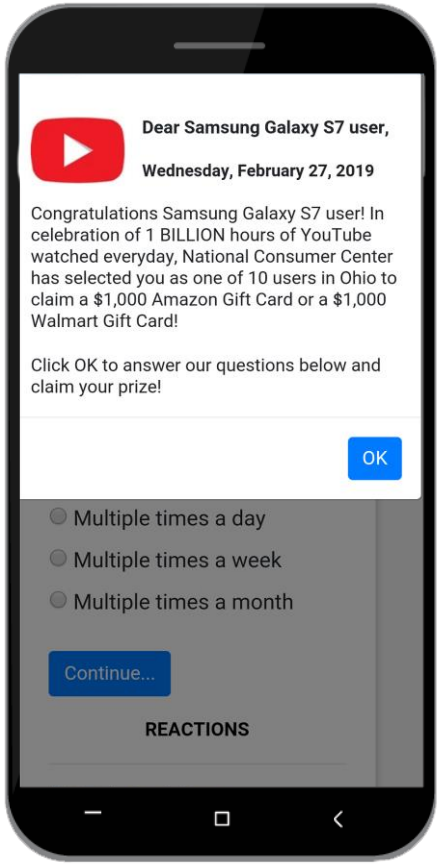
Attack surfaces are extremely reduced!

Security issues_

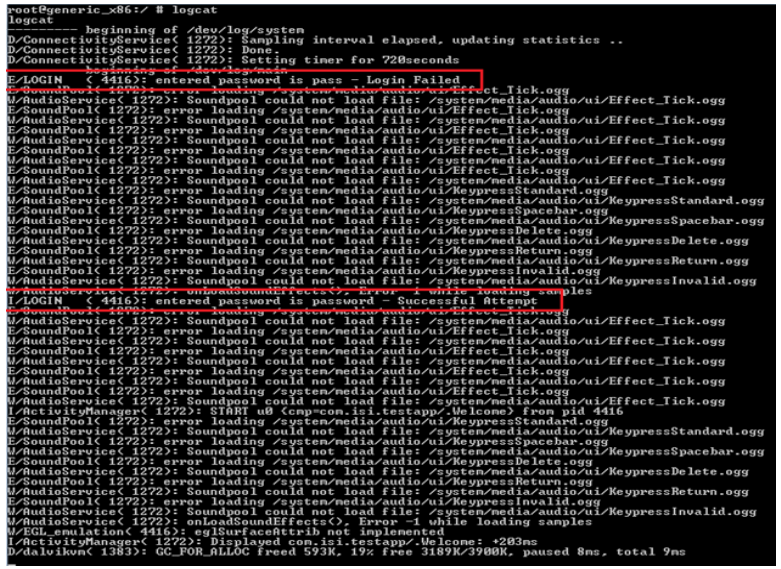
User is the weakest link







Developers often make mistakes



Too much data in logs

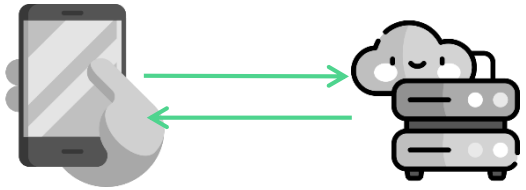
```
package com.august.util;

import android.content.SharedPreferences;

public class Settings
{
  private static final String ENC_KEY = "[redacted]";
  private static final LogUtil LOG = LogUtil.getLogger(Settings.class);
  public static final String SIZE_SUFFIX = "*size*";
  public static final String STR_ACCESS_TOKEN = "API_ACCESS_TOKEN";
  public static final String STR_DEBUG_SETTINGS = "DEBUG_SETTINGS";
  public static final String STR_INSTALL_TOKEN = "API_INSTALL_TOKEN";
  public static final String STR_PUSH_ALERTS = "PUSH_ALERTS";
  public static final String VERSION_SUFFIX = "_v1";
  static Settings _instance = null;
  DebugSettings _debugSettings = new DebugSettings();
  Properties _encryptedProps = null;

  public static Settings init()
  {
    if (_instance == null) {
      _instance = new Settings();
```

Hardcoded keys



Insecure data transfer

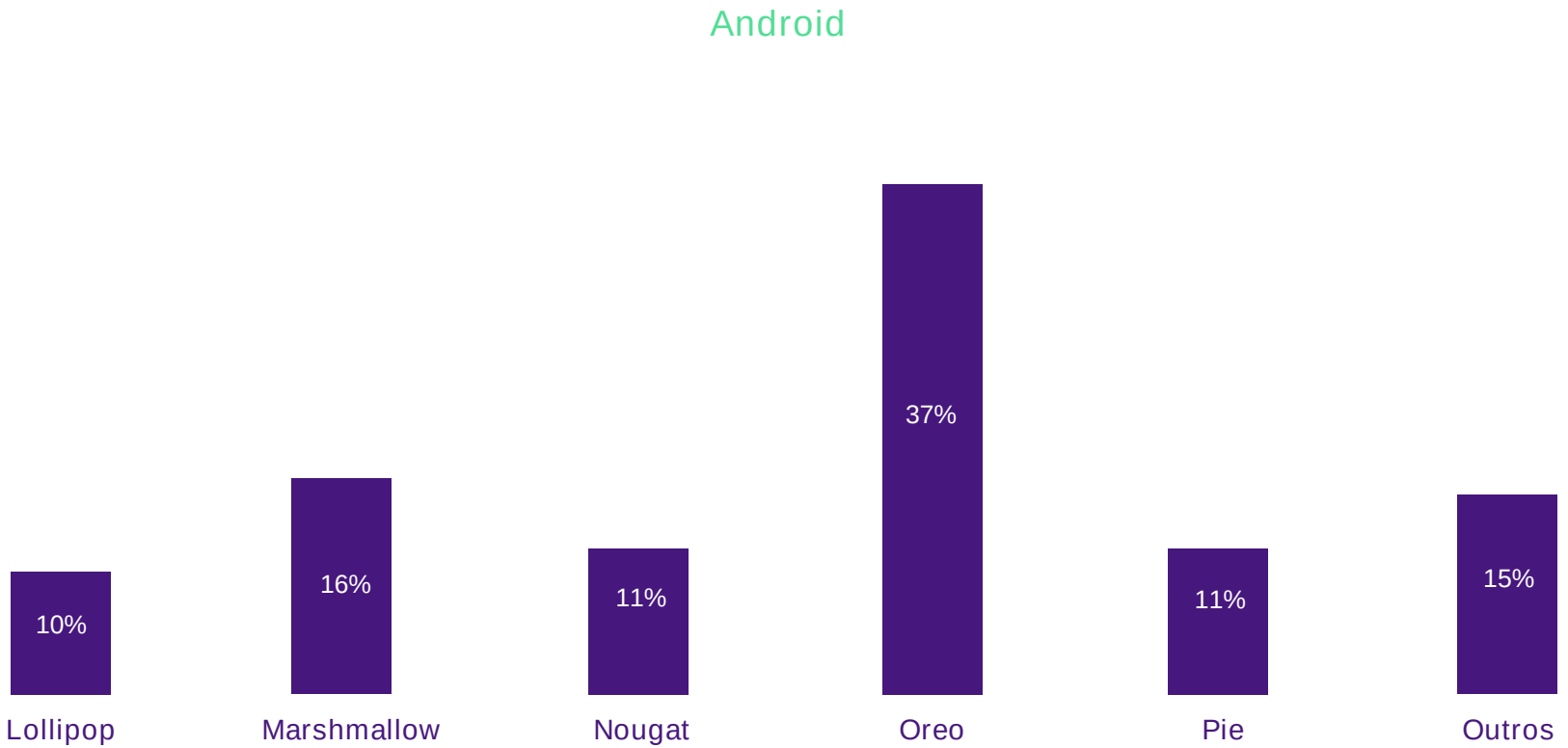
Insecure storage

Android is safe, not perfect.

782

Number of critical vulnerabilities found on Android between 2011 and 2018

Updates are necessary!



BONUS:

How Brazilian authorities were hacked?