



Operating Systems

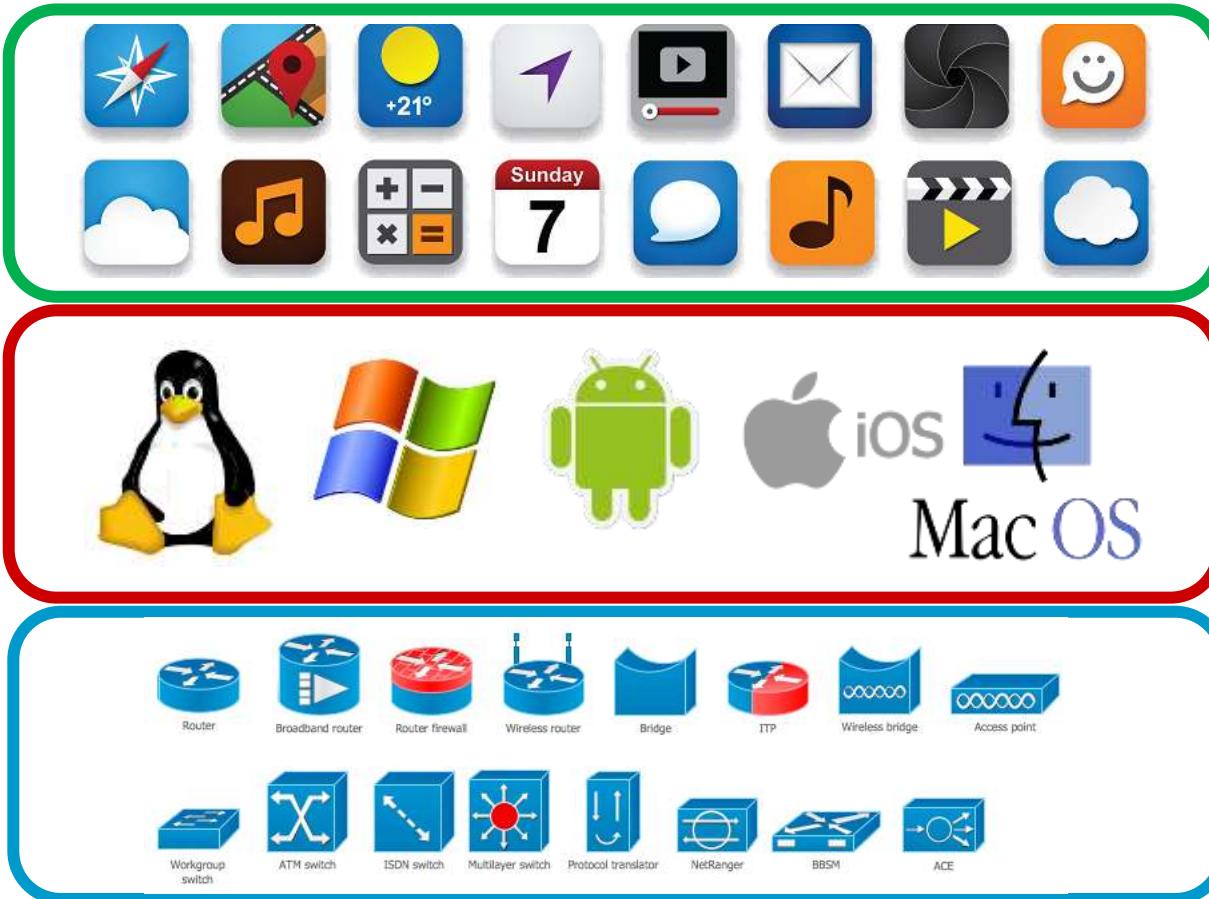


Michał Szychowiak, PhD

<https://www.cs.put.poznan.pl/mszychowiak/en>



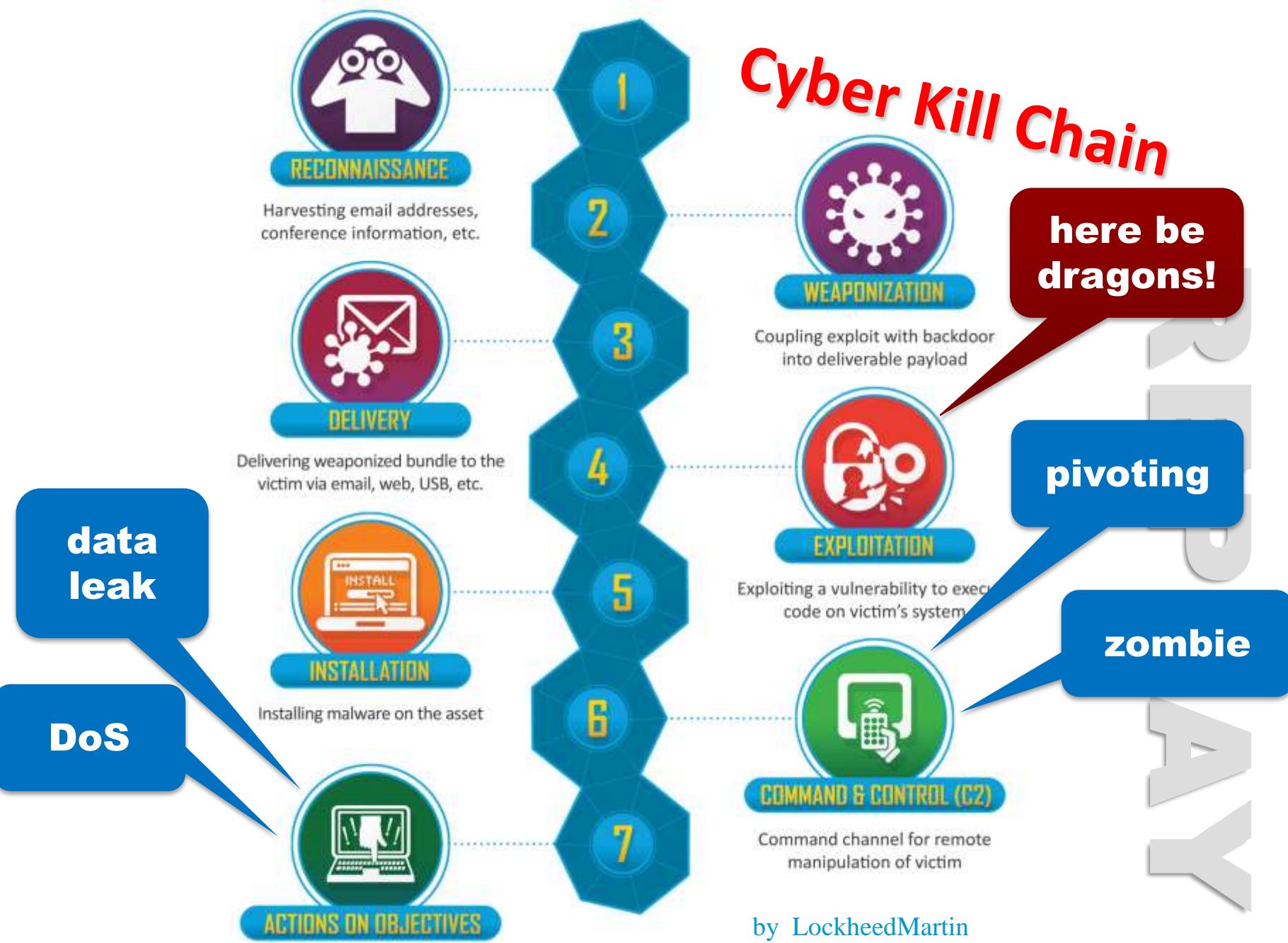
3 Tiers Architecture



Agenda

- 1. OS Vulnerabilities**
- 2. Defense**
 1. Authentication
 2. Authorization & Access Control
- 3. Malware infections**
- 4. Covered channels**

Cyber Kill Chain

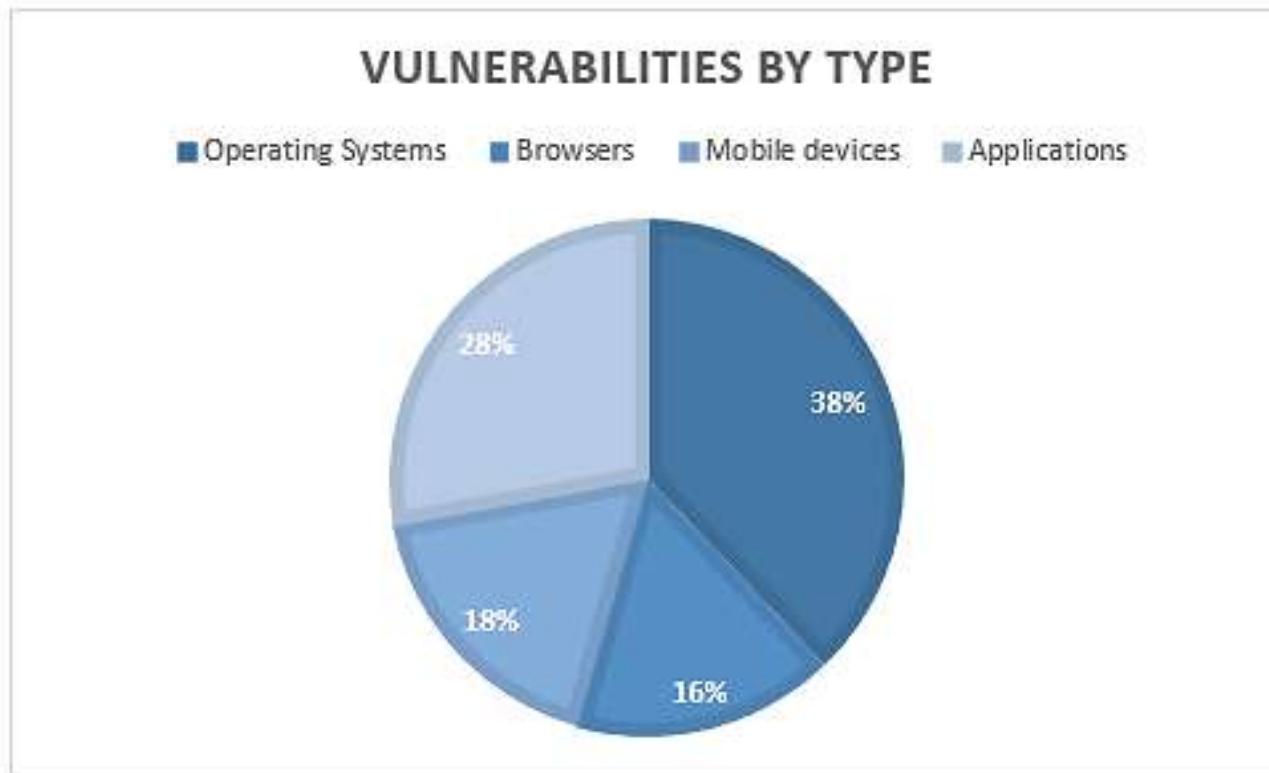


by LockheedMartin

Vulnerabilities

Common Vulnerabilities and Exposures (CVE) database

<https://cve.mitre.org>





SOLARIS

 ReactOS



 **TinyOS**



Mac OS



 **RIOT**

QUBES OS

A REASONABLY SECURE OPERATING SYSTEM



FreeBSD®



 BlackBerry

MeeGo

 **VxWorks**

OpenWrt
Wireless Freedom

 **CISCO**
IOS

 **MikroTik**
RouterOS

TIZEN *

QNX

 **Obada**



Firefox OS

NUCLEUS




Cisco IOS
Cisco NX-OS
Cisco IOS-XR

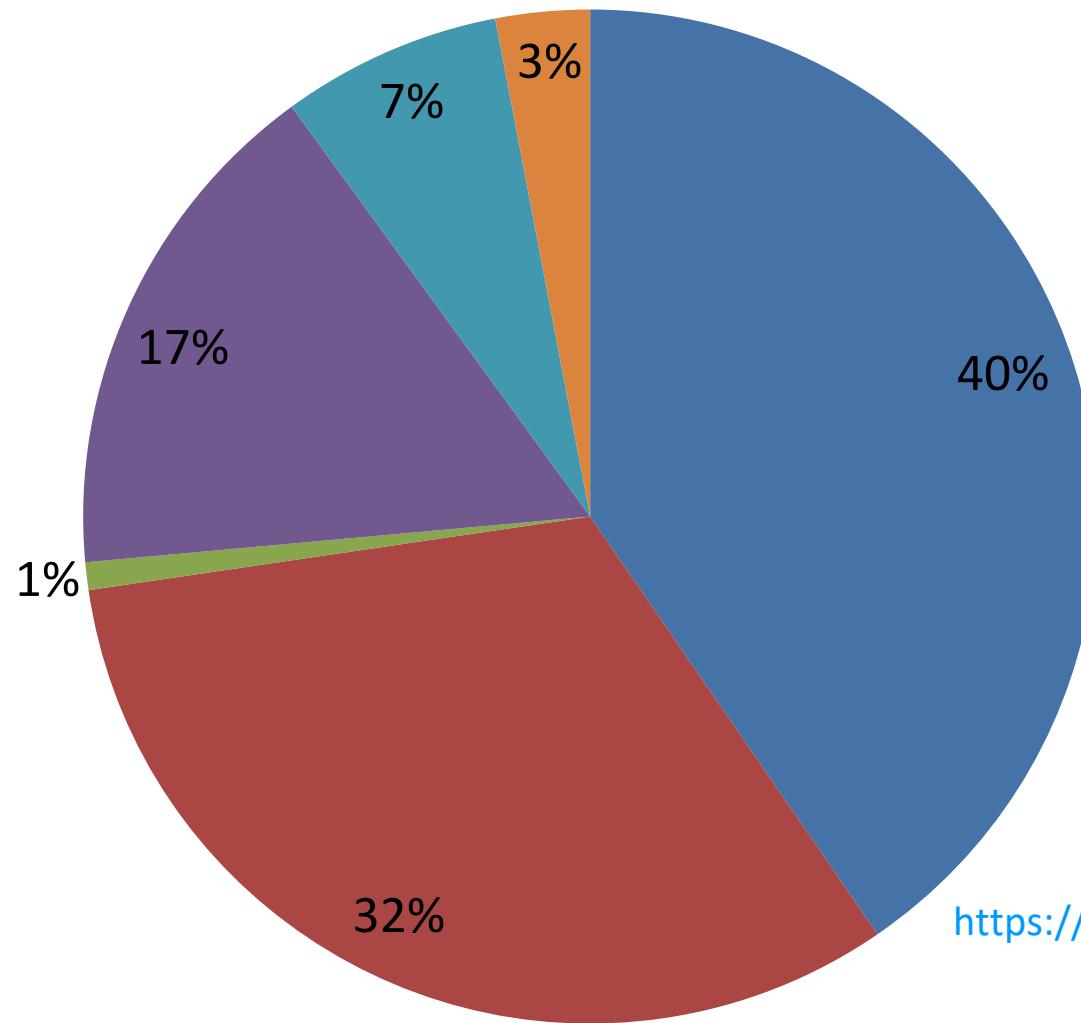
 **ROS**

 Open Source Robotics Foundation



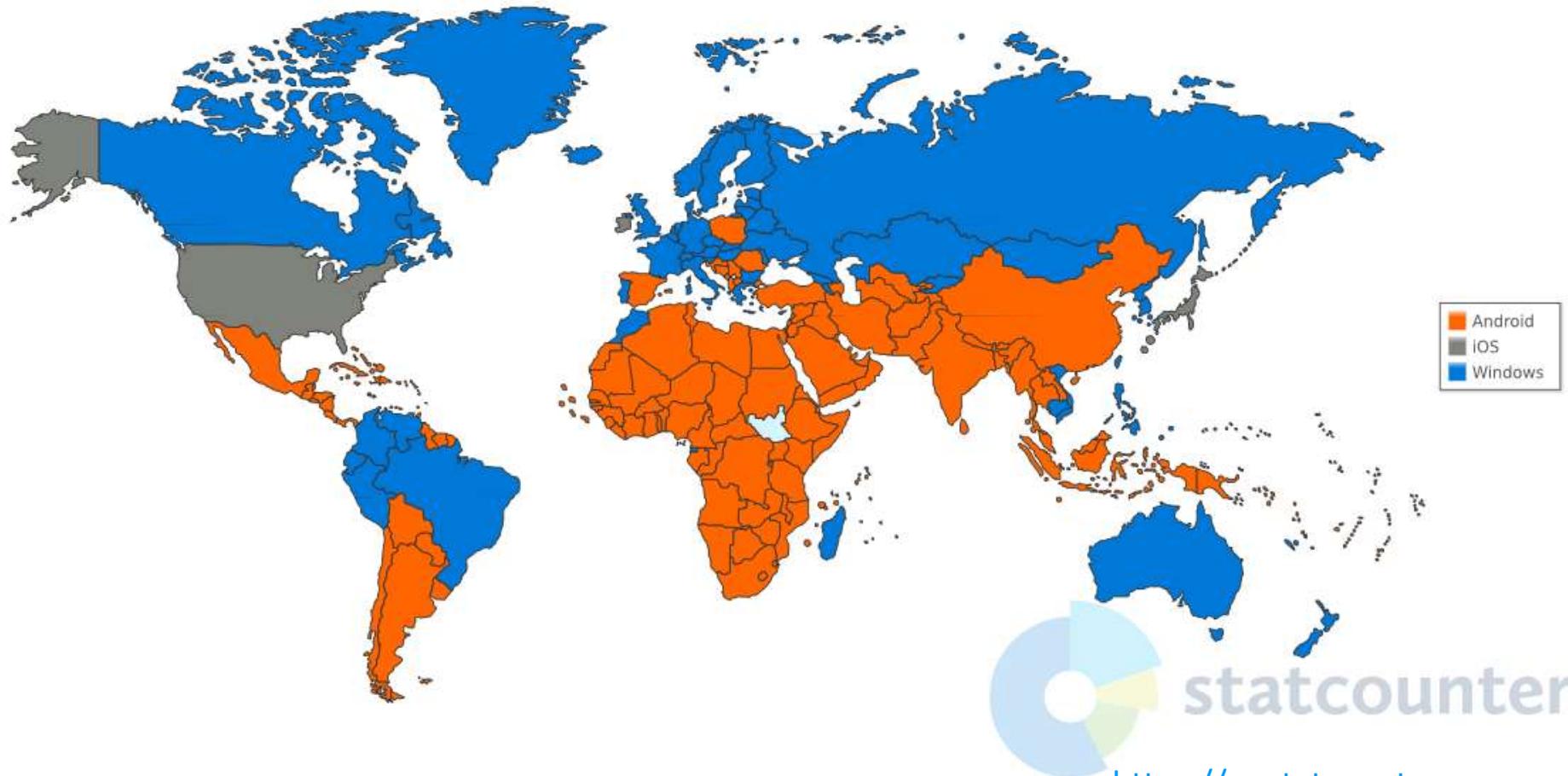
OS Market Share

■ Android ■ Windows ■ Linux ■ iOS ■ OS X ■ Other



<https://gs.statcounter.com>

StatCounter Global Stats
Desktop, Mobile & Tablet Operating System Market Share Worldwide from Feb - Apr 2021



OS fingerprinting

Remote fingerprinting

- ➔ Active
 - ➔ initiating interactions with OS components (services)
- ➔ Passive
 - ➔ eavesdropping existing interactions

OS fingerprinting

Remote fingerprinting

→ Banners

```
GET / HTTP/1.1
Host: www.example.com
User-Agent: Mozilla/5.0 (Windows NT 10; x64) Firefox/85.0
Accept: text/html,application/xhtml+xml,application/xml
Accept-Language: pl,en-US,en
Accept-Encoding: gzip, deflate
Connection: keep-alive
...
...
```

OS fingerprinting

Remote fingerprinting

- ➔ OS differentiating features

Test type	Usable Protocol	Test precision
Directory Separator	HTTP	Windows vs. Unix
New line characters	HTTP	Windows vs. Unix
Special/reserved filenames	HTTP	Windows vs. Unix
Root directory	FTP	Windows, Unix, Symbian, OS/2
Special characters (EOF,EOL)	-	-
Filesystem limitations	HTTP, FTP	Correlates FS-type to OS
Filesystem illegal characters	HTTP, FTP	Correlates FS-type to OS
Case sensitivity	HTTP, FTP	Windows vs. Unix
Special filenames handling	HTTP, FTP	Windows vs. Unix
Special files in directory	HTTP, FTP	Windows types, MacOS, Unix

OS fingerprinting

Remote fingerprinting

- RTO (Retransmission Time-Out)

MacOS X	FreeBsd	Linux	Win
SYNACK	SYNACK	SYNACK	SYNACK
SYNACK	SYNACK	SYNACK	SYNACK
SYNACK	SYNACK	SYNACK	SYNACK
SYNACK	SYNACK	SYNACK	
SYNACK		SYNACK	
RST		SYNACK	



AUTHENTICATION

Authentication



Unix

IEEE 1003.1 (POSIX)

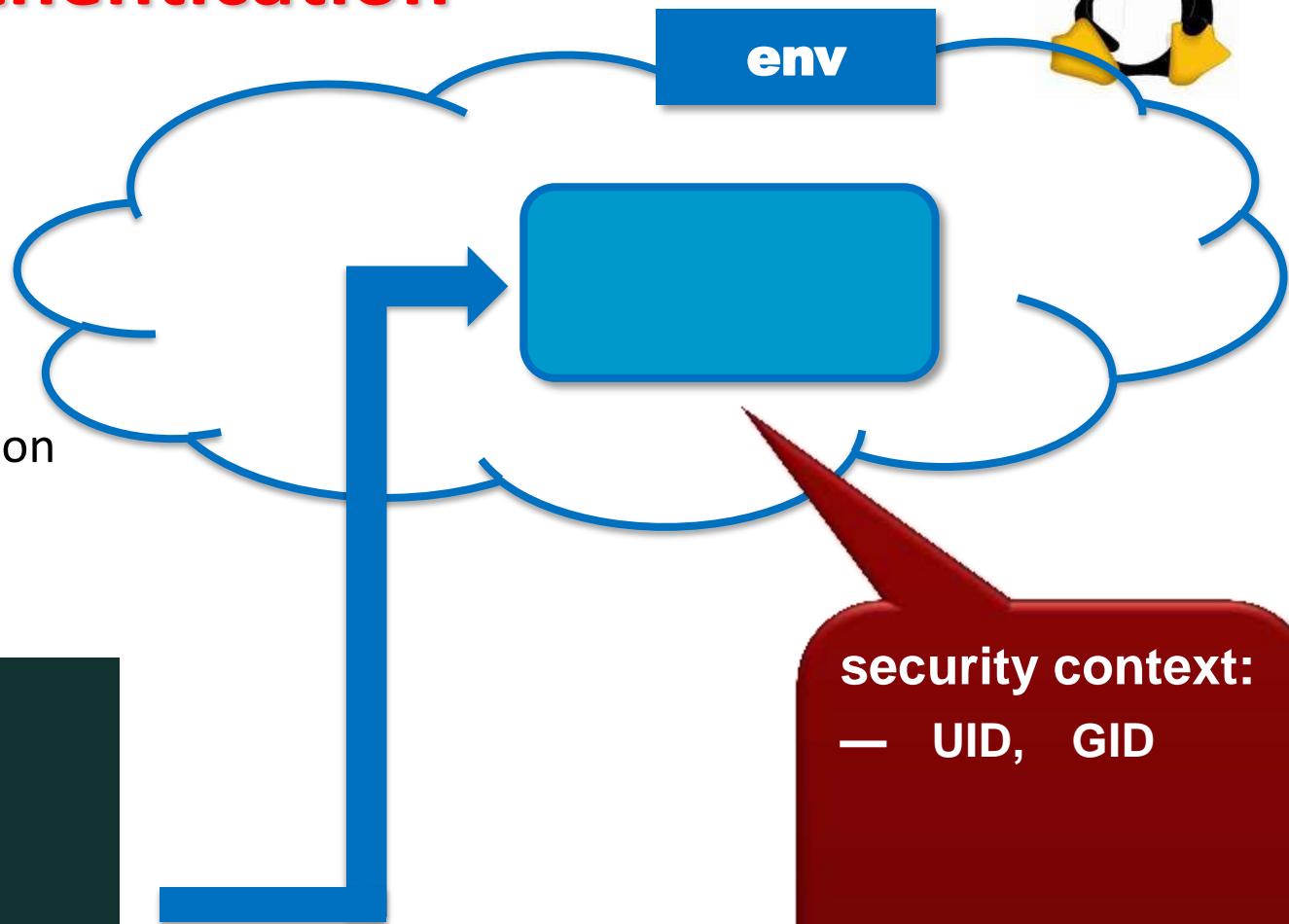
Single Unix Specification

login



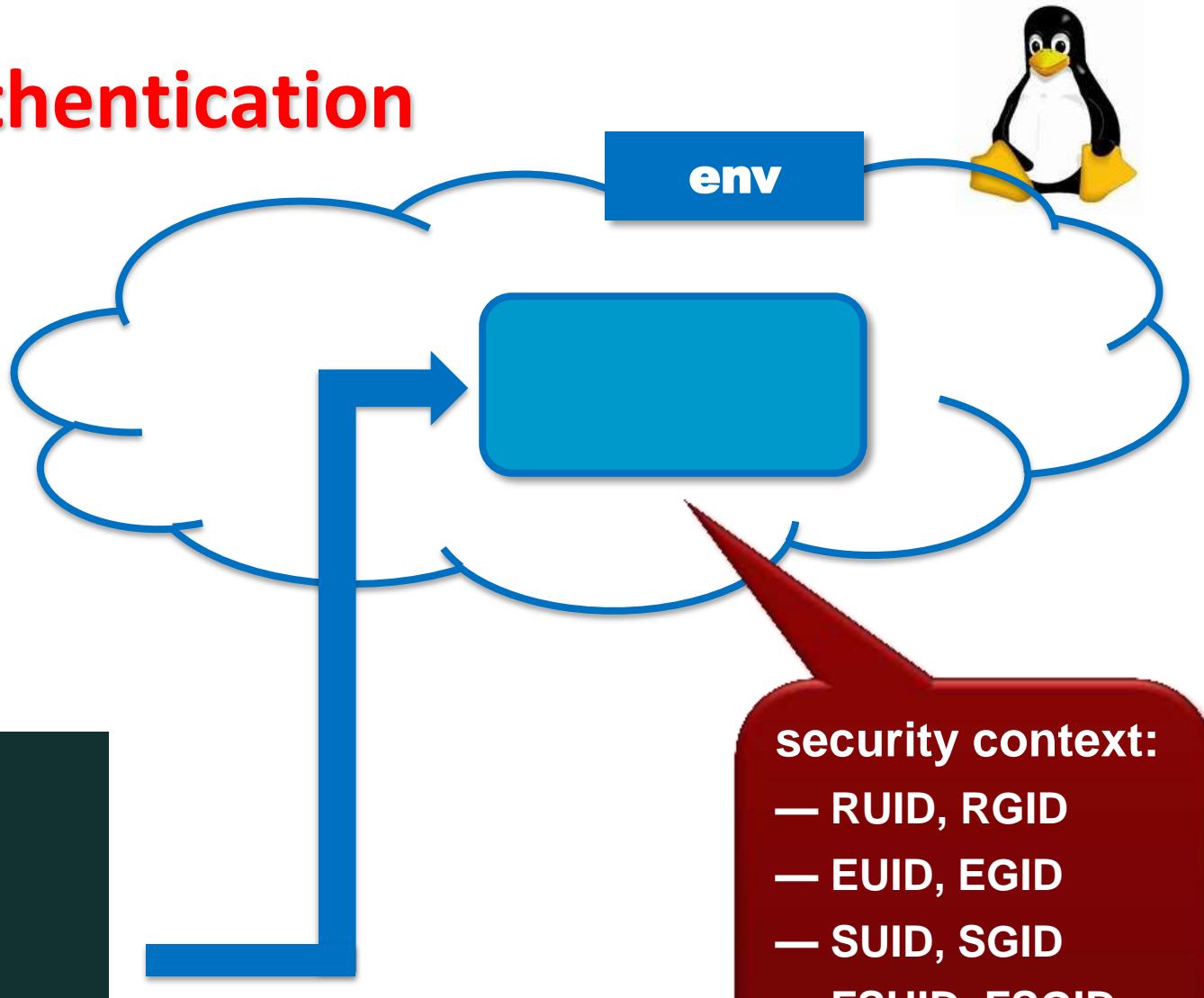
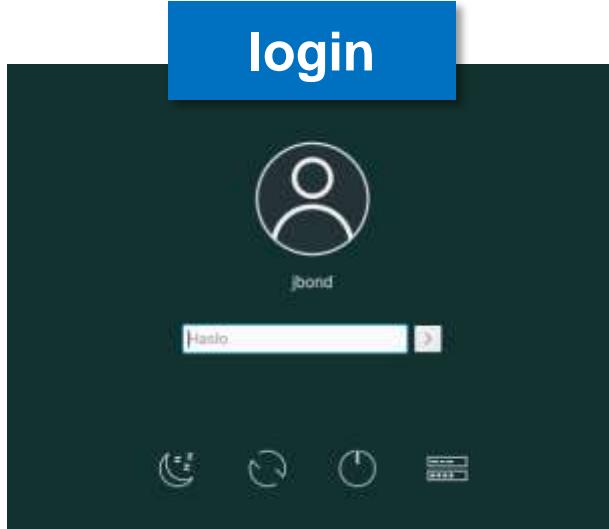
jbond

haslo:



Authentication

Unix



security context:

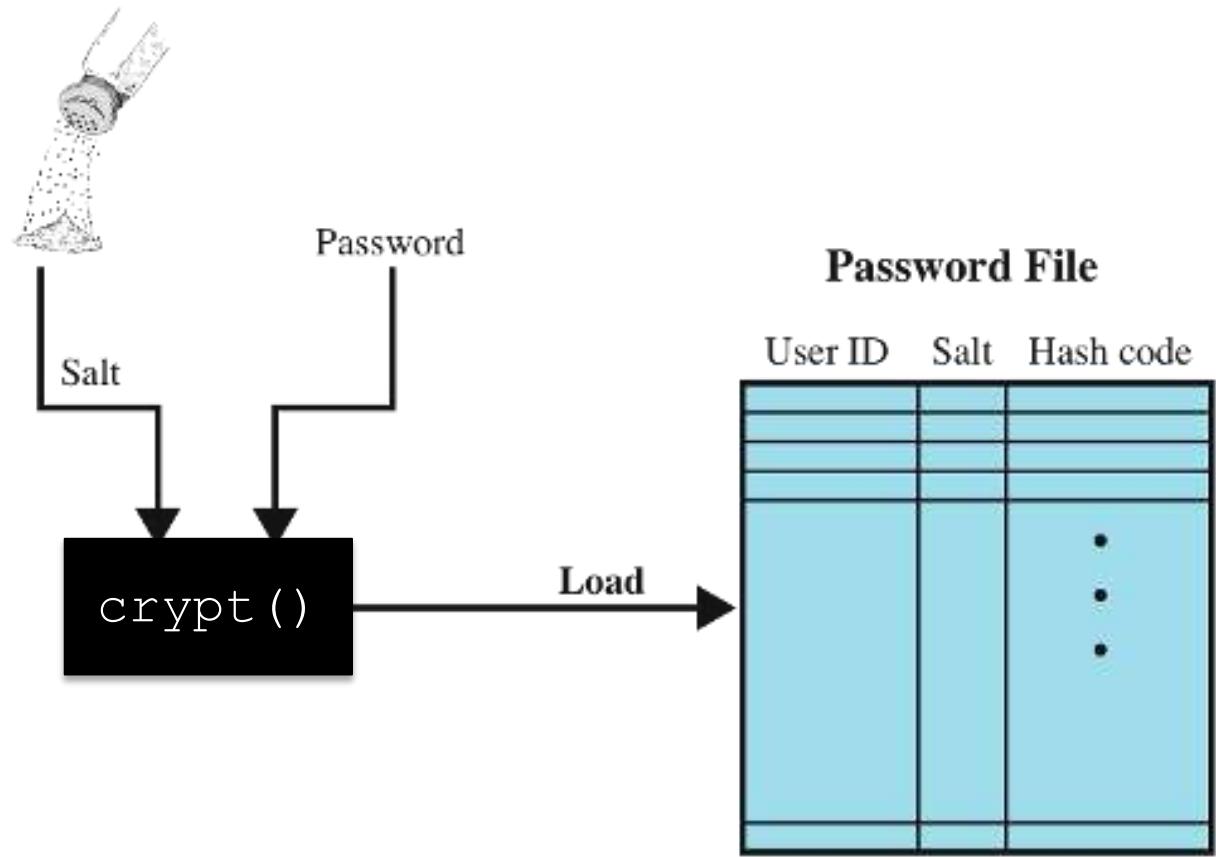
- RUID, RGID
- EUID, EGID
- SUID, SGID
- FSUID, FSGID
- ...

Authentication



Unix

- /etc/passwd
- crypt()





Authentication

Unix

- ➔ /etc/passwd
- ➔ crypt()

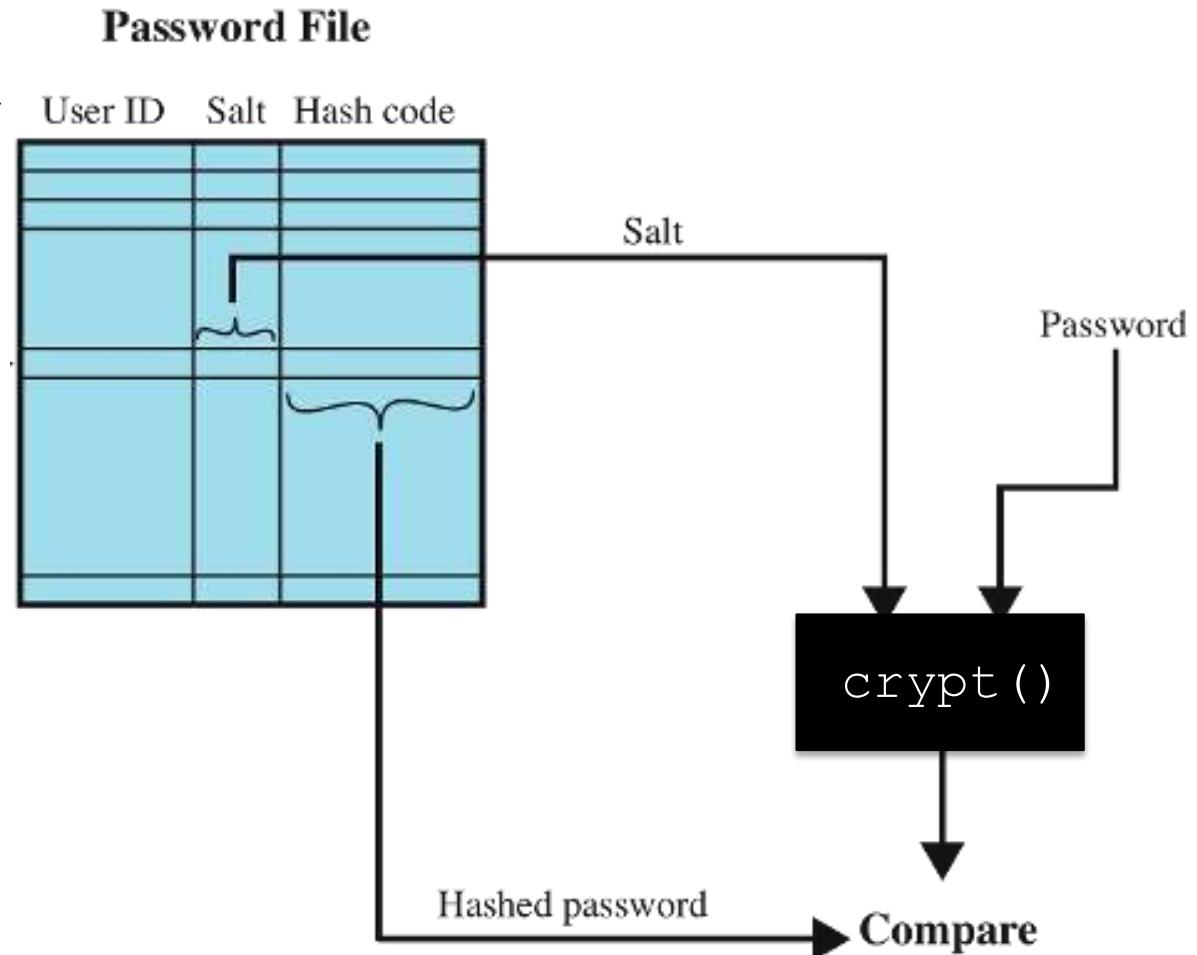
```
wnj:ZDjXDBwXle2gc:8:2:Bill Joy,457E,7780:/a/guest/wnj:/bin/csh
dmr:AiInt5qKdjmHs:9:2:Dennis Ritchie:/a/guest/dmr:
ken:sq5UDrPlKj1nA:10:2:& Thompson:/a/guest/ken:
mike:KnKNwMkyCt8ZI:11:2:mike karels:/a/guest/mike:/bin/csh
carl:S2KiTfS3pH3kg:12:2:& Smith,508-21E,6258:/a/guest/carl:/bin/csh
joshua::999:2:&:/usr/games:/usr/games/wargames
```

Authentication



Unix

- /etc/passwd
- crypt()





Authentication

Unix

- /etc/passwd → /etc/shadow → NIS, NetInfo, LDAP
- crypt() → SHA-2, bcrypt(), scrypt(), ...





Authentication

Unix

- /etc/passwd → /etc/shadow → NIS, NetInfo, LDAP
- crypt() → SHA-2, bcrypt(), scrypt(), ...

Hash stretching:

CRYPT=SHA512

SHA512_CRYPT_FILES=500000

/etc/security/crypt.conf



Authentication

Unix

YaST2 — YaST2 - security @ vlinux

Security Overview
Predefined Security Configurations
Password Settings
Boot Settings
Login Settings
User Addition
Miscellaneous Settings

Password Settings

Checks

Check New Passwords

Minimum Acceptable Password Length
8

Number of Passwords to Remember
12

Password Encryption Method
SHA-512

Password Age

Minimum	Maximum
2	60

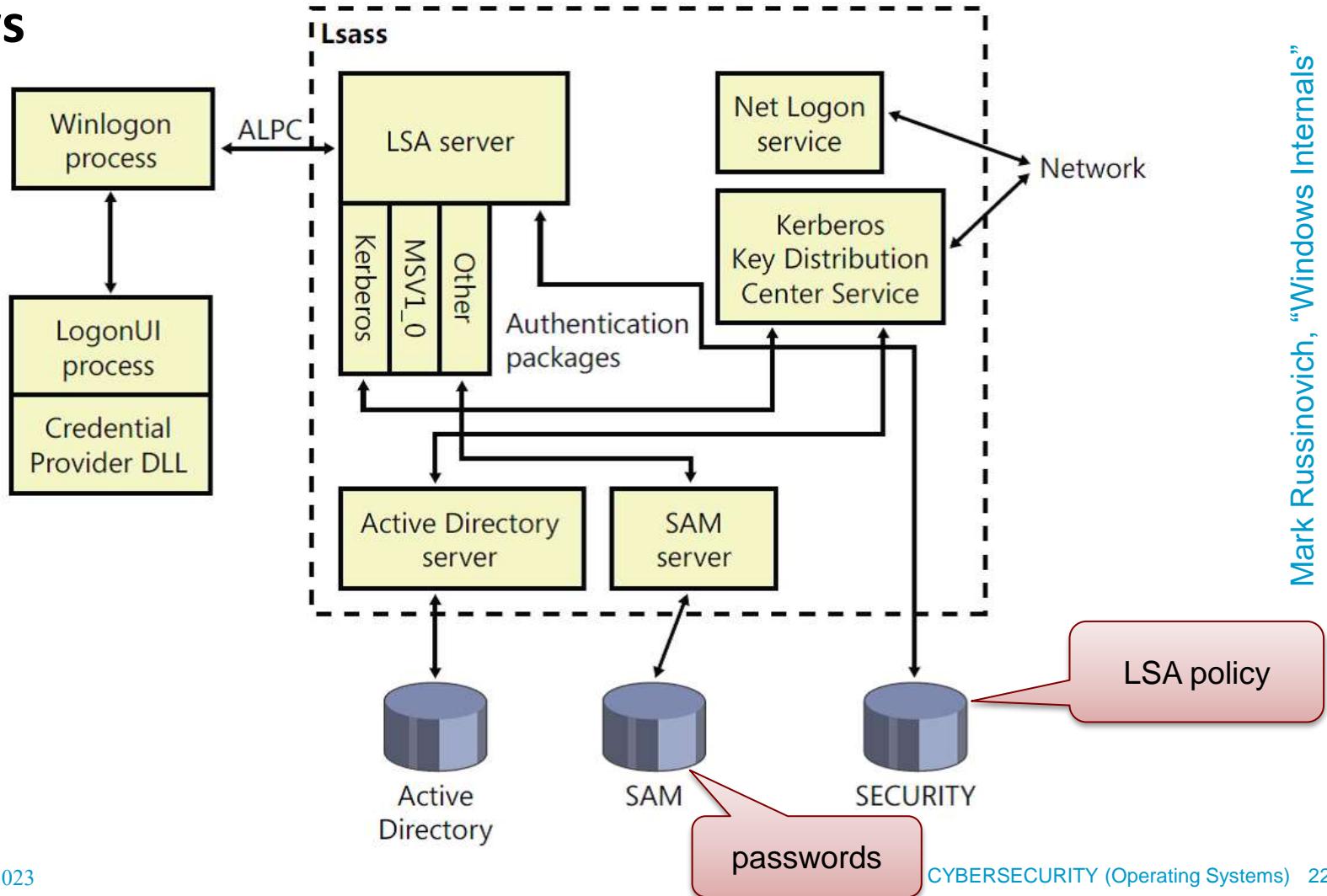
Days before Password Expires Warning
7

Help Cancel **OK**

Authentication



Windows



Authentication



Windows

- ➔ SAM (Security Accounts Manager)
NTFS: %SYSTEMROOT%\SYSTEM32\CONFIG\SAM ⇒ HKLM\SAM
- ➔ NTLM hash (variant of MD4) 128b
- ➔ storage encrypted with RC4 → AES-128 SysKey
- ➔ Win7: Password-Based Key Derivation Function (PBKDF) + SHA-256
- ➔ Win10: Windows Hello (biometrics) + Credentials Guard (→VBS)
- ➔ Active Directory: Kerberos



HOMEWORK

=

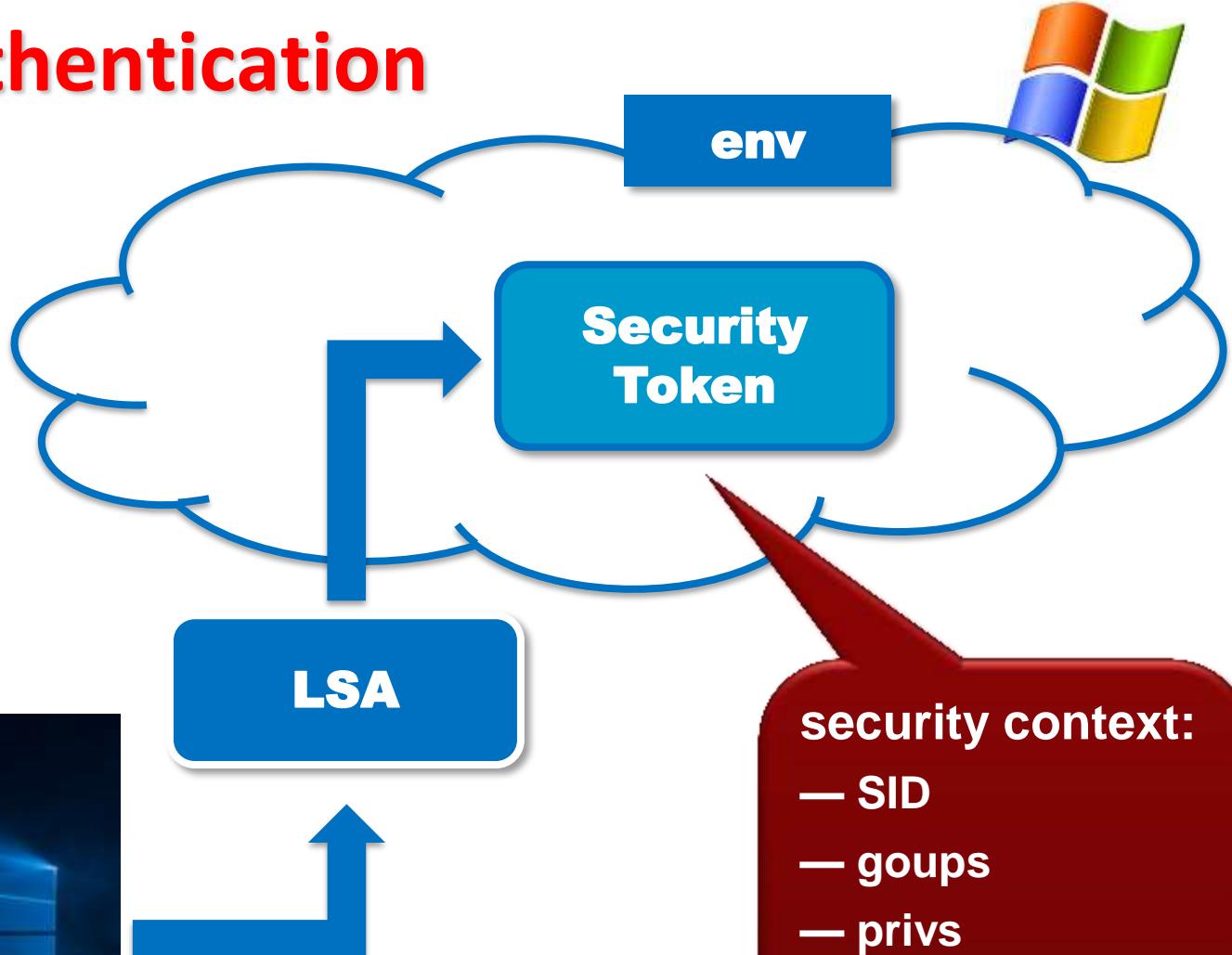
Half Of My Energy Wasted On Random Knowledge



→ Pass the Hash (PtH) attack

Authentication

Windows

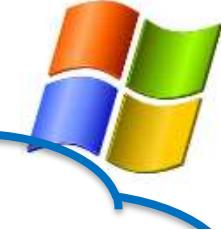


security context:

- SID
- groups
- privs
- integrity level
- ...

Authentication

env

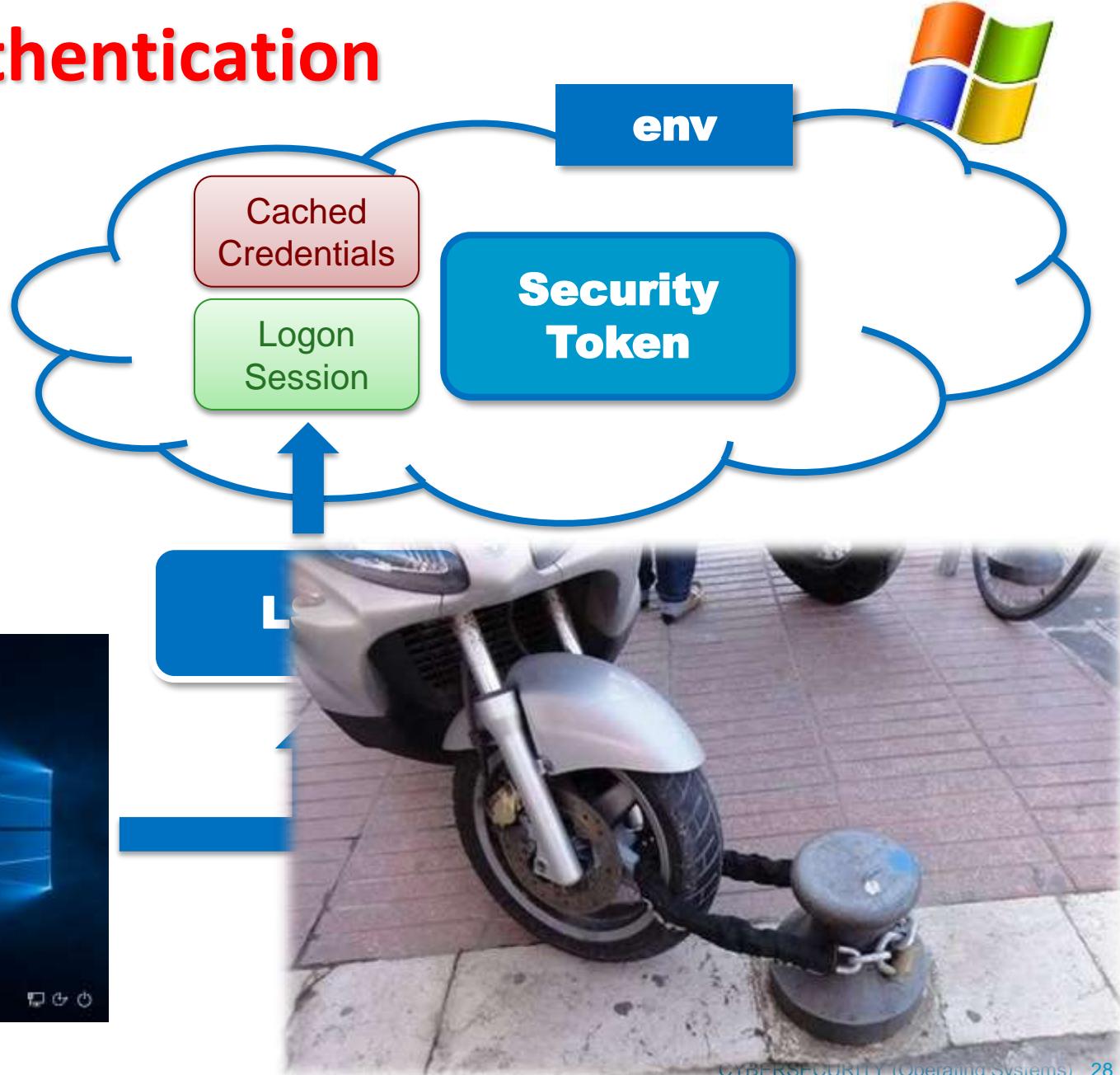


```
PS C:\Users\Administrator> $token = Get-NtToken -User JamesBond  
PS C:\Users\Administrator> Show-NtToken $token  
PS C:\Users\Administrator>
```

Main Details	Groups	Privileges	Default Dacl	Misc	Operations	Security
User VIRT-WIN\JamesBond - TokenId 00000000-00204933						X
						—
Name	Flags					
SeBackupPrivilege	Disabled					
SeChangeNotifyPrivilege	Default Enabled					
SeCreateGlobalPrivilege	Default Enabled					
SeCreatePageFilePrivilege	Default Enabled					
SeCreateSymbolicLinkPrivilege	Default Enabled					
SeDebugPrivilege	Disabled					
SeDelegateSessionUserImpersonatePrivilege	Disabled					
SeImpersonatePrivilege	Disabled					
SeIncreaseBasePriorityPrivilege	Default Enabled					
SeIncreaseQuotaPrivilege	Default Enabled					
SeIncreaseWorkingSetPrivilege	Default Enabled					
SeLoadDriverPrivilege	Disabled					
SeManageVolumePrivilege	Default Enabled					
SeProfileSingleProcessPrivilege	Default Enabled					
SeRemoteShutdownPrivilege	Default Enabled					
SeRestorePrivilege	Disabled					
SeSecurityPrivilege	Default Enabled					
SeShutdownPrivilege	Default Enabled					
SeSystemEnvironmentPrivilege	Default Enabled					
SeSystemProfilePrivilege	Default Enabled					
SeSystemTimePrivilege	Default Enabled					
SeTakeOwnershipPrivilege	Disabled					
SeTimeZonePrivilege	Default Enabled					
SeUndockPrivilege	Default Enabled					

Authentication

Windows



Authentication



security context:

— ...

— privs:

1) ...

2) ...

3) ...

4) ...

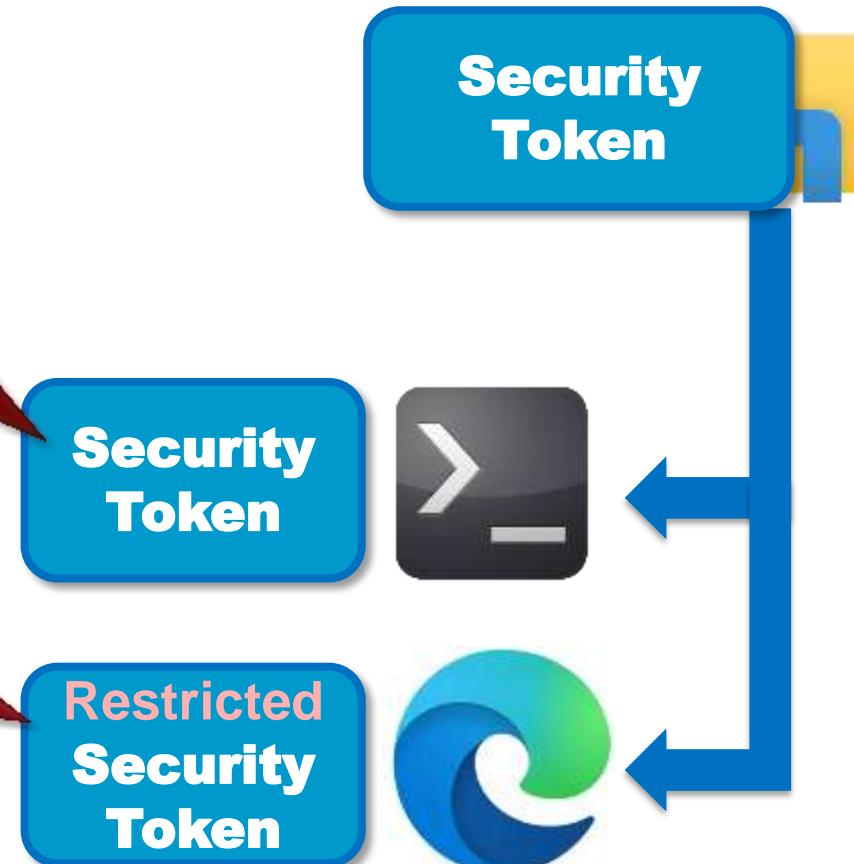
security context:

— ...

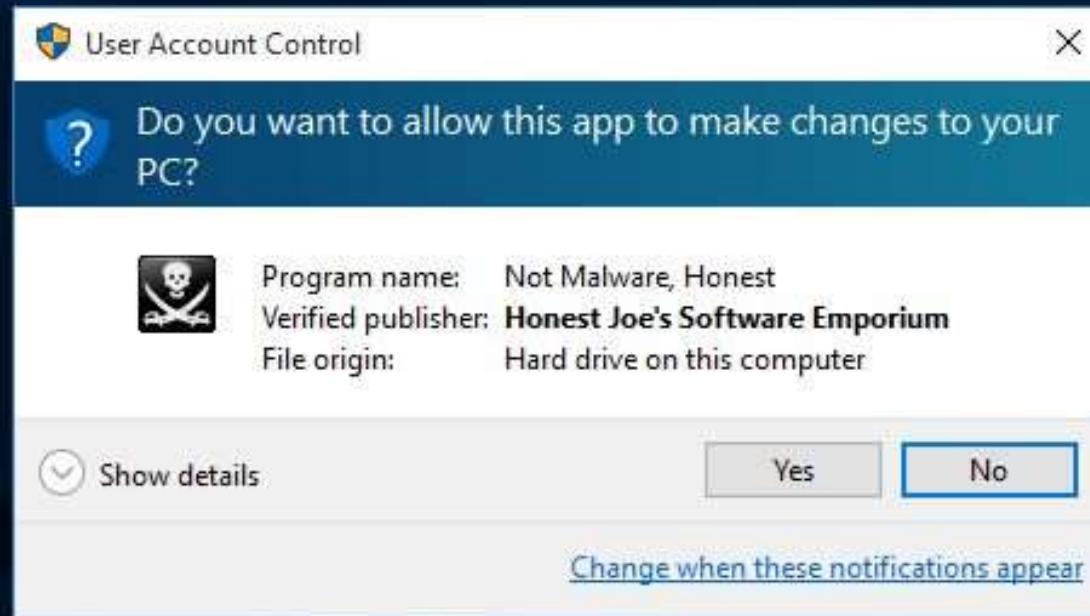
— privs:

2) ...

4) ...



User Account Control (UAC)





Authentication



Windows

➔ credwiz.exe

← 🔎 Stored User Names and Passwords X

Back up or restore your stored user names and passwords

Back up your stored user names and passwords:
If your user names and passwords are lost, damaged, or destroyed then you can use this backup to restore them. You can also use this backup to transfer your user names and passwords to other computers.

Restore your stored user names and passwords

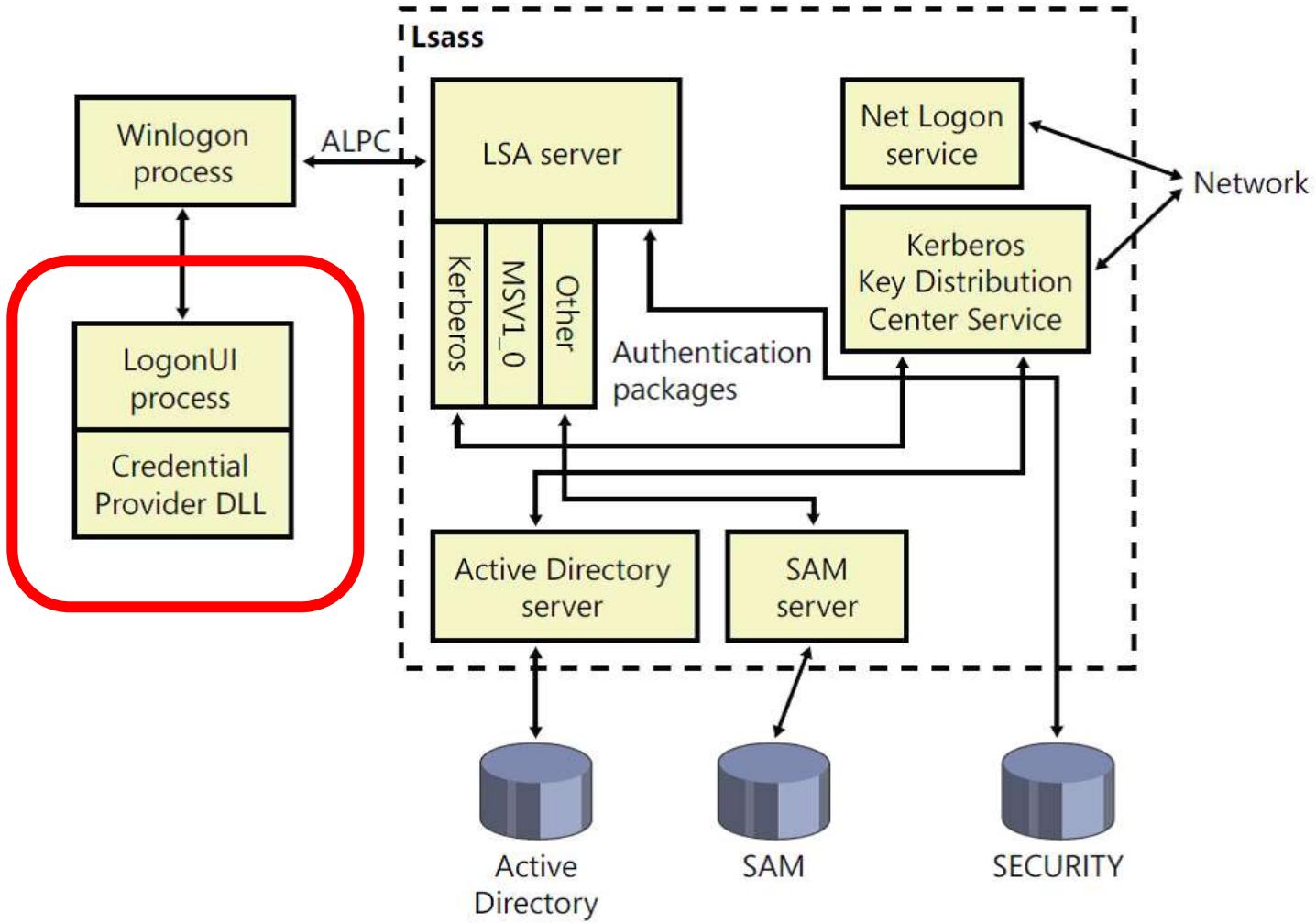
[Next](#) [Cancel](#)



Authentication

Windows

REPLAY

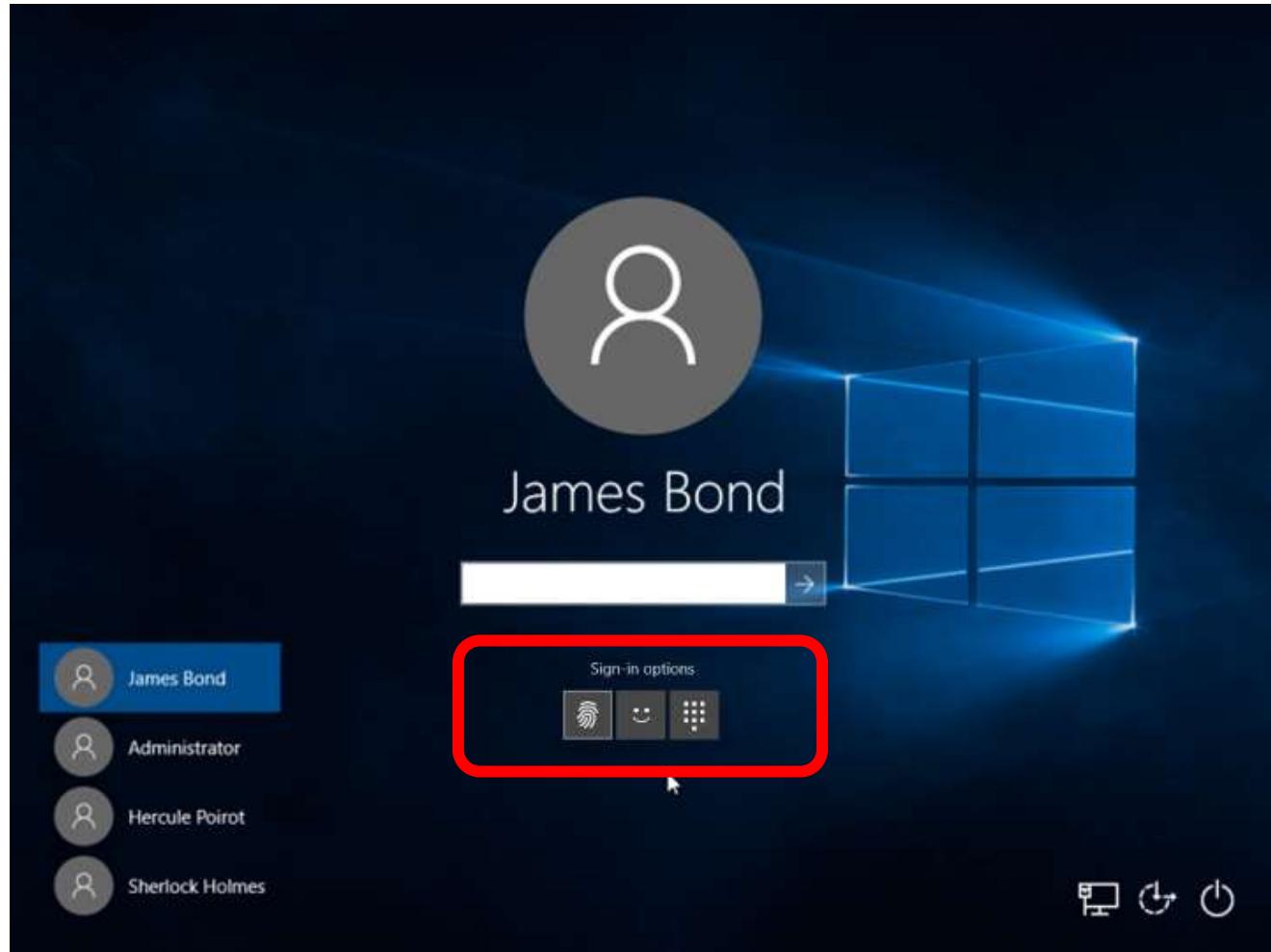


Mark Russinovich, "Windows Internals"

Authentication



Windows



Authentication

One-Time Passwords

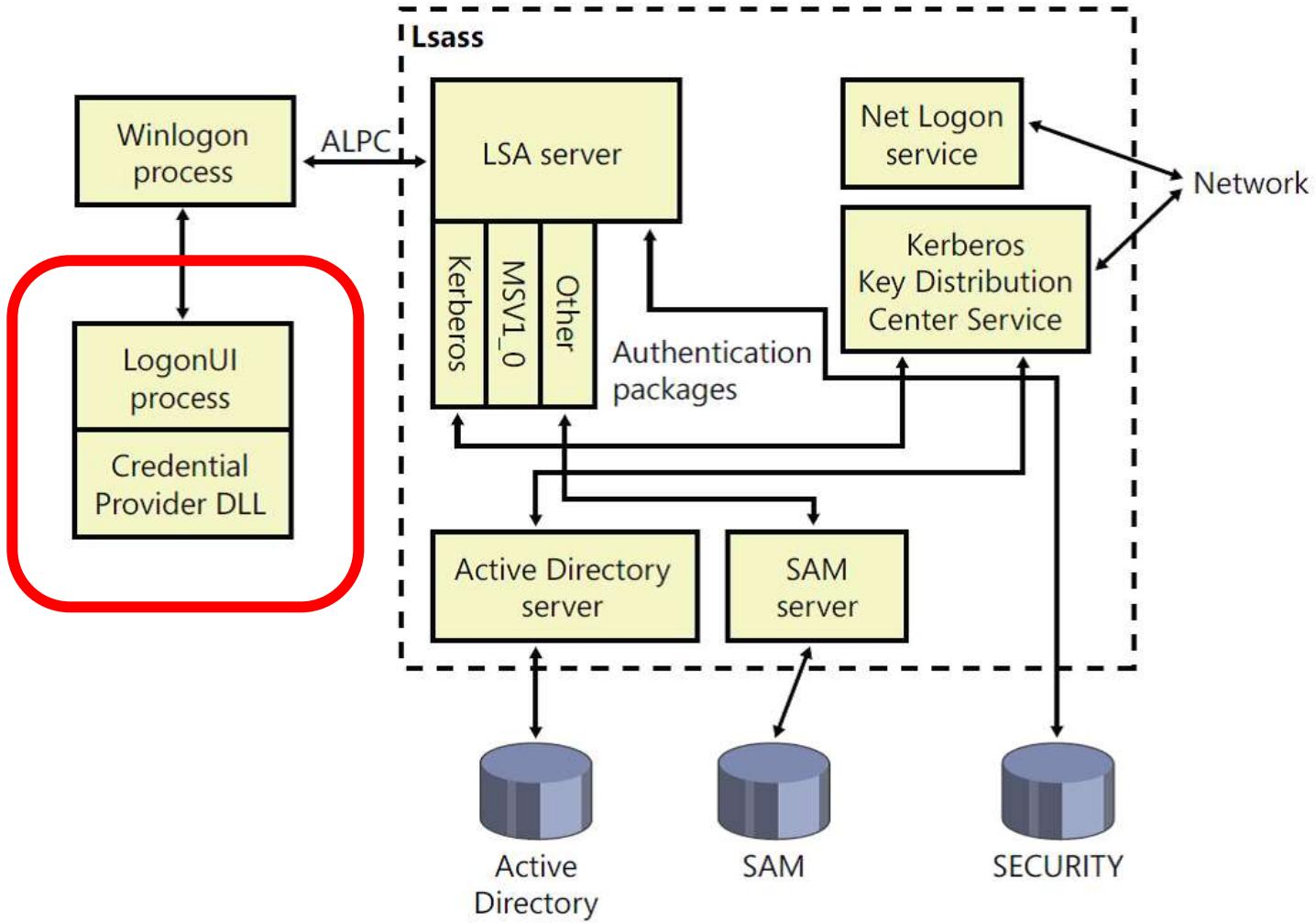




Authentication

Windows

REPLAY



Mark Russinovich, "Windows Internals"

Authentication

One-Time Passwords



Authentication

One-Time Passwords

Hardware:



Software:

S/Key

OPIE (One-time Passwords In Everything)

HOTP (HMAC OTP):

- open standard (RFC 4226)
- Time-based OTP (TOTP , RFC 6238)

...

Authentication

One-Time Passwords

S/Key (Lamport '80)

- no crypto involved, only a hash function
- initial password: P
- hash list: $P, h(P), h(h(P)), \dots, h^{t-1}(P)$
- i -th password = $h^{t-i}(P)$, $1 \leq i \leq t$
- initial password reuse: $h^{t-i}(P|salt)$

Authentication

Two-Factor Authentication (2FA)

Multi-Factor Authentication (MFA)



Authentication



1. Local passwords

- ➔ hashes
- ➔ hash stretching and protection
- ➔ password policy

2. OTP, 2FA (MFA)

3. Security tokens (→ authorization)

- ➔ impersonation
- ➔ Consent & Credentials (UAC)

AUTHORIZATION

&

ACCESS CONTROL

Authorization & Access Control



Unix — POSIX (*Portable Operating System Interface*)

POSIX 1003.1

- r w x
- u g o
- SUID, SGID, sticky

POSIX 1003.1e/1003.2c

- ACL (also NFSv3, but not NFSv4!)
- CAP
- MAC: Trusted Solaris, Trusted IRIX, TrustedBSD, ...
- LO MAC (Low Water-Mark MAC): Linux, FreeBSB

Authorization & Access Control



POSIX ACL

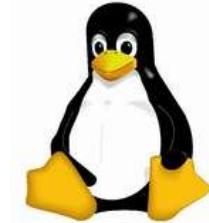
Access Control Entry

- base ACE = u g o
- extended ACE:

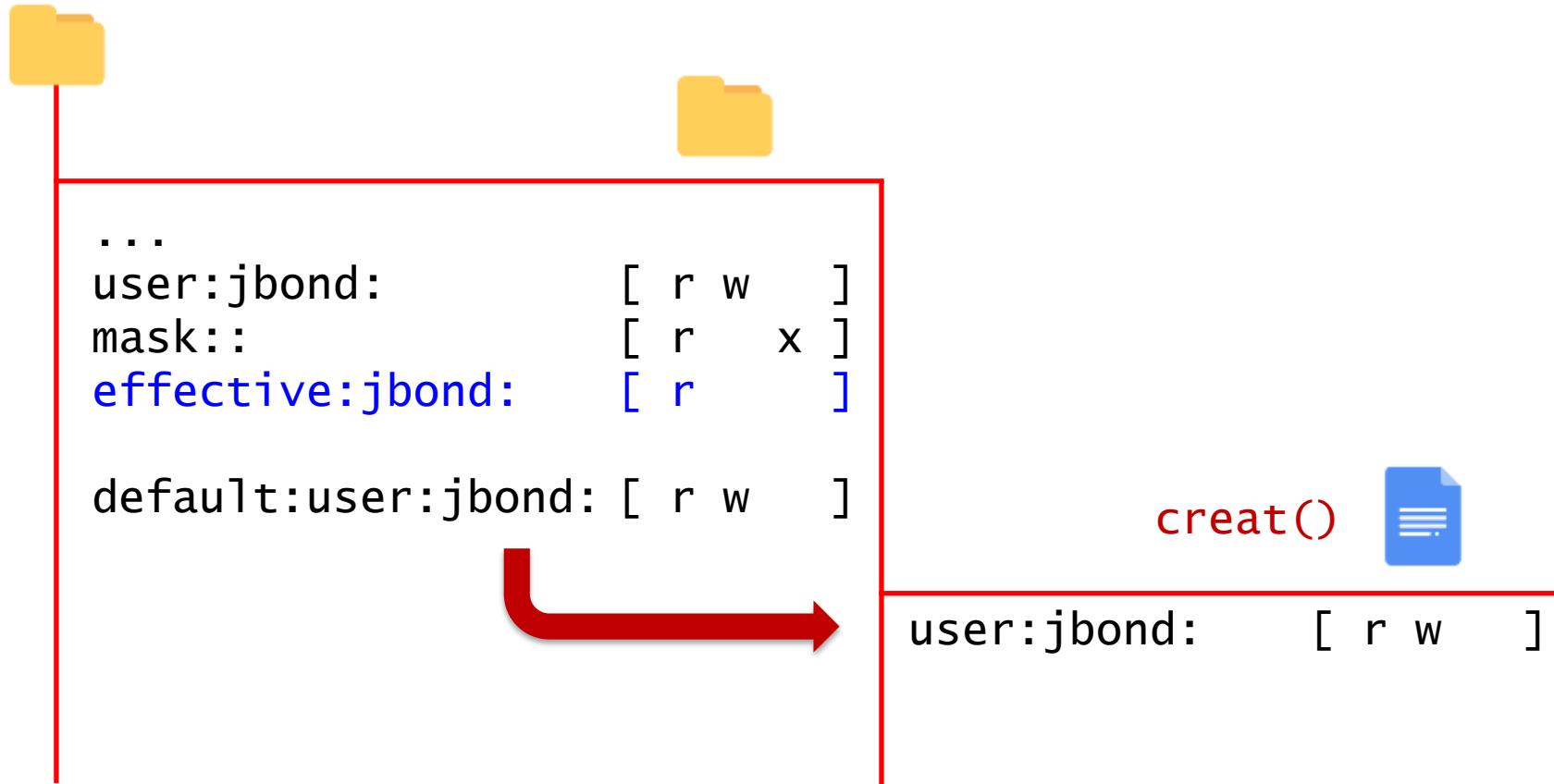


user::	[r w x]
user:jbond:	[r w]
group::	[r]
group:agents:	[r x]
other::	[]
mask::	[r x]
effective:jbond:	[r]

Authorization & Access Control



POSIX ACL





Authorization & Access Control

POSIX ACL

```
$ setfacl -n -m user:jbond:rwx .
$ getfacl --omit-header .
user::rwx
user:jbond:rwx
group::r-x
group:agents:r-x
mask::r-x
other::---
default:user::rwx
default:group::r-x
default:mask::r--
default:other::---
```

Authorization & Access Control

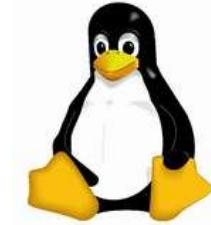


Trustees

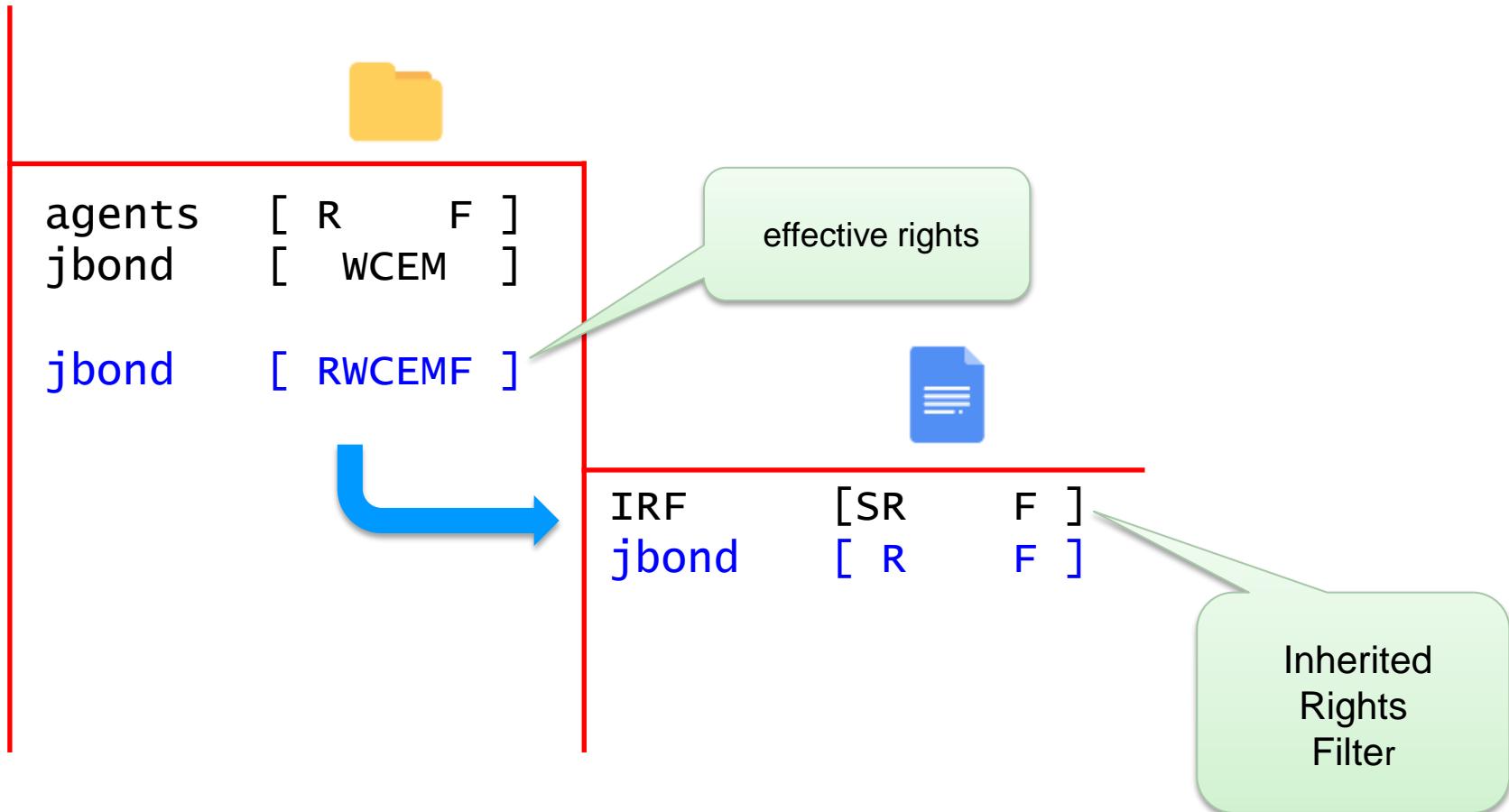
- NetWare (Novell), Linux (<http://trustees.sourceforge.net/>), RSBAC

R (Read)
W (Write)
C (Create)
E (Erase)
M (Modify)
F (File scan)
A (Access control)
S (Supervisory)

Authorization & Access Control



Trustees



Authorization & Access Control



Windows

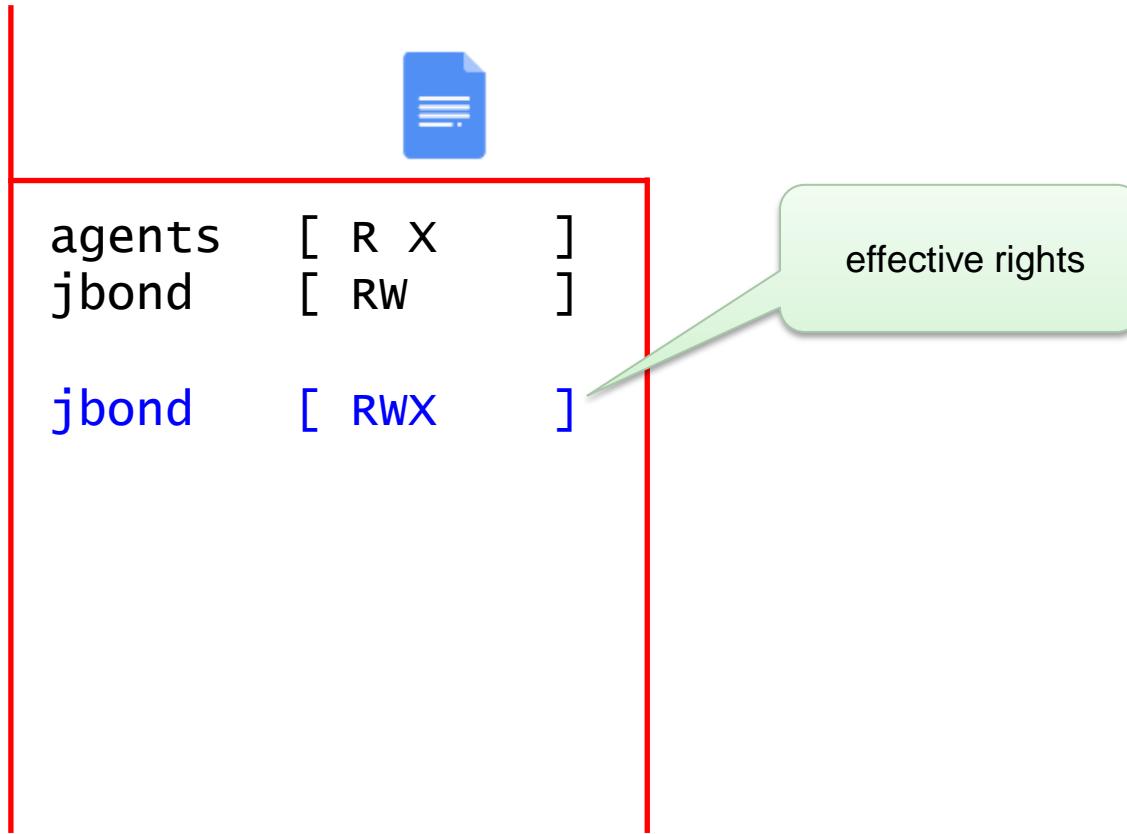
- ACL in NTFS (called DACL)
- SMB/CIFS (Common Internet File System) support
- distinct GRANT and DENY assignments
- dynamic groups (such as All Authenticated Users)
- multiple attributes: append
delete
change permissions
take ownership
...

similar to chattr , richACL, and NTFSv4 for Unix

Authorization & Access Control



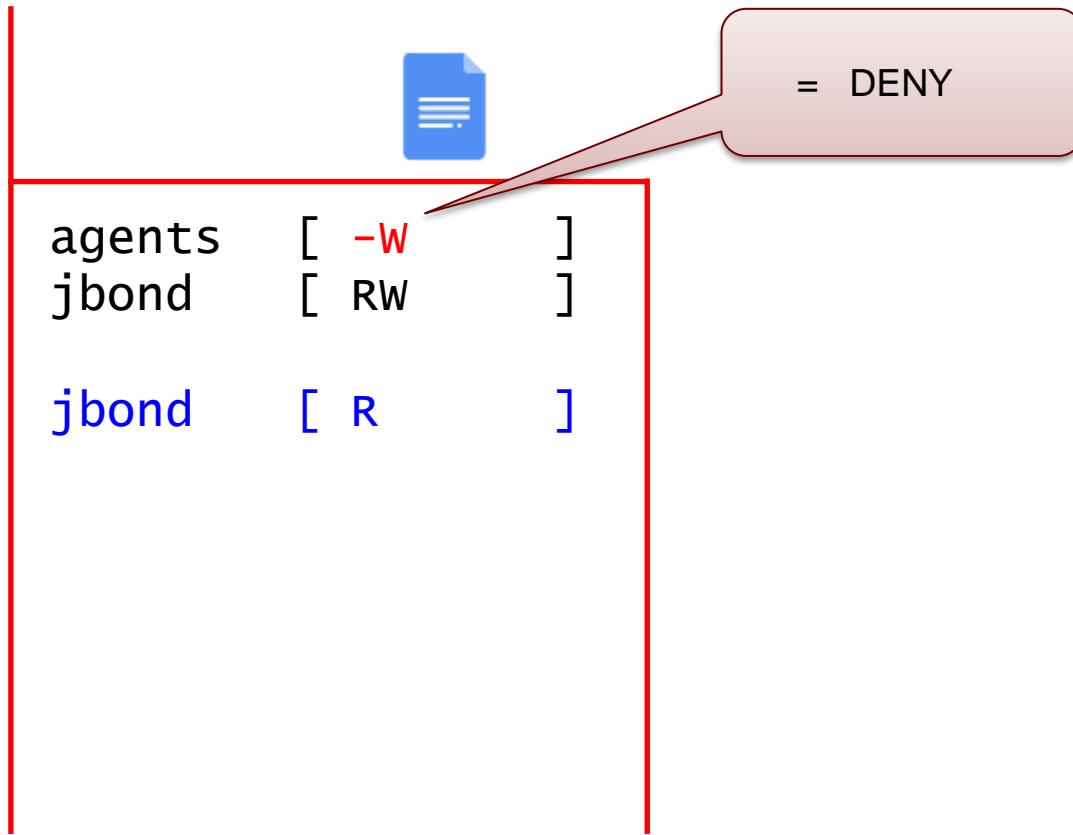
Windows DACL



Authorization & Access Control



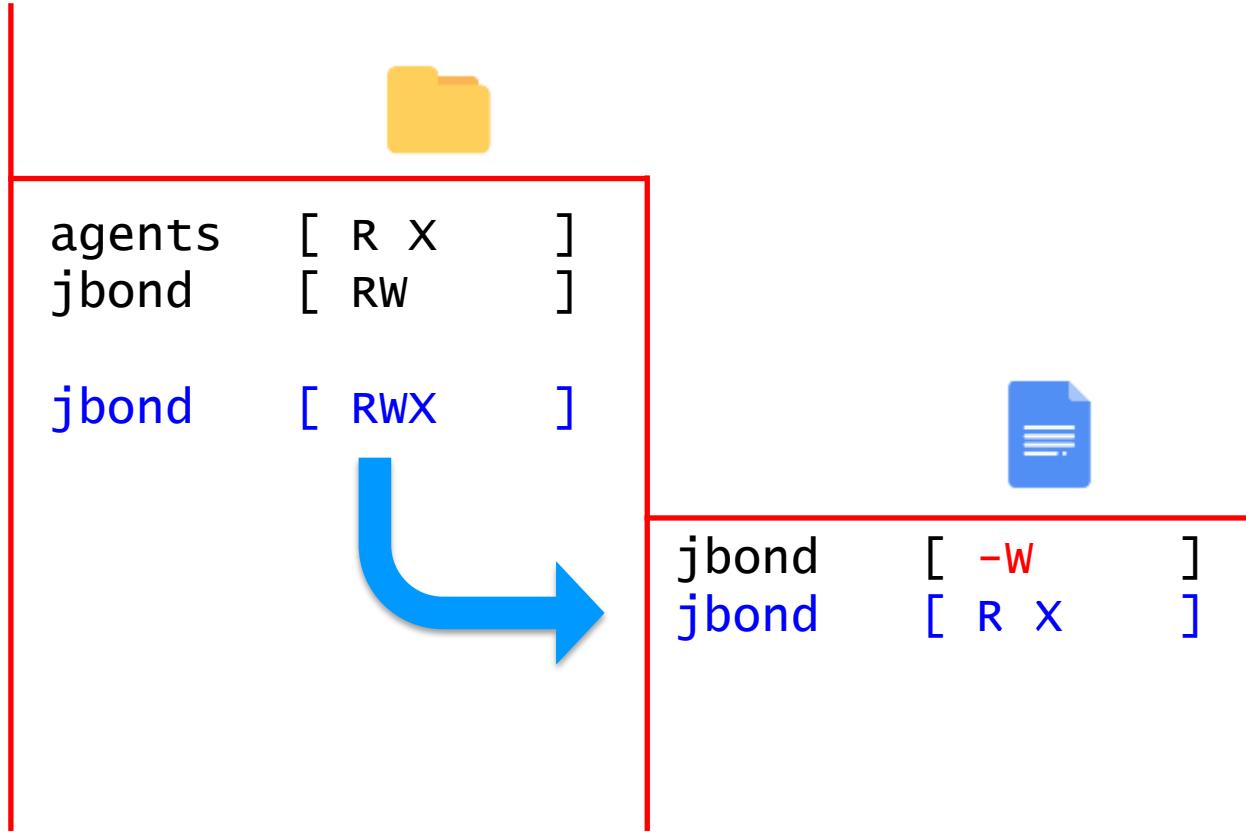
Windows DACL



Authorization & Access Control



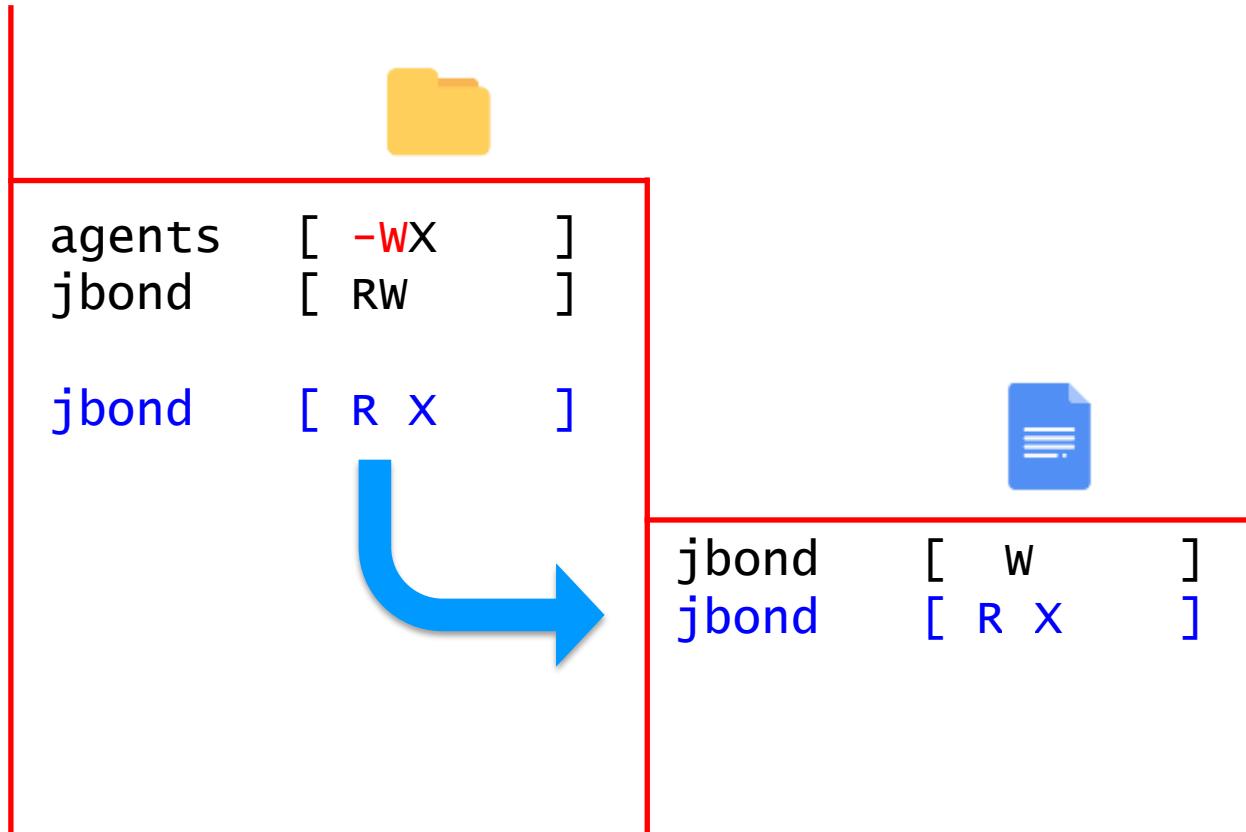
Windows DACL



Authorization & Access Control



Windows DACL



Mandatory Integrity Control (MIC)



Windows Integrity Levels

0 = untrusted: SID S-1-16-0x0000

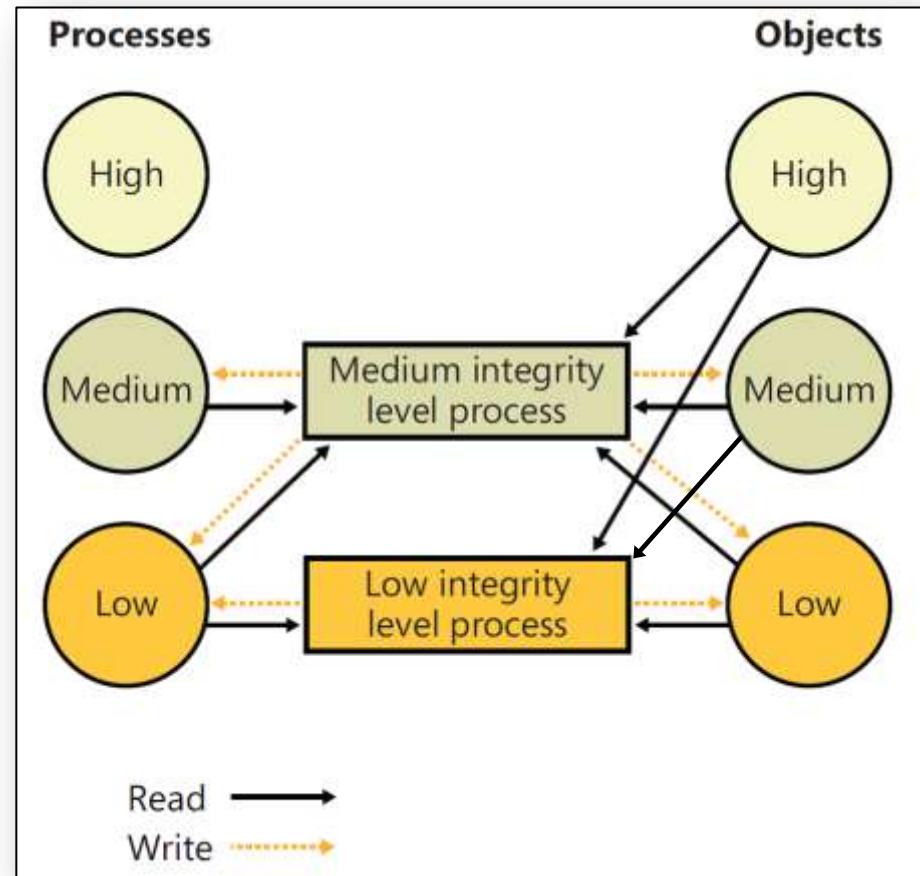
1 = low: SID S-1-16-0x1000

2 = medium: SID S-1-16-0x2000

3 = high: SID S-1-16-0x3000

4 = system: SID S-1-16-0x4000

5 = protected: SID S-1-16-0x5000





Authorization & Access Control

Android

```
shell@android:/ $ ps
USER     PID   PPID  VSIZE  RSS   PC NAME
root      1     0    640    496   S /init
root     46     1   4660   1200   S /system/bin/vold
root     48     1   9772   1268   S /system/bin/netd
system   371    52  307064  46084   S system_server
shell    410    52  233948  17132   S adbd
...
radio    520    52  259604  25716   S com.android.phone
u0_a7    524    52  255172  45060   S com.android.systemui
u0_a8    534    52  248952  56996   S com.android.launcher
u0_a16   789    52  244992  20612   S com.android.calendar
u0_a59   819    52  246240  20104   S com.bar.foo
...
```



Authorization & Access Control

Android

```
shell@android:/ $ ls -ld /data/data/com.bar.foo  
drwxr-x--x u0_a59 u0_a59 2018-02-27 14:01 com.bar.foo
```



Authorization & Access Control

Android

- system user

```
shell@android:/ $ id system
uid=1000(system) gid=1000(system)
groups=1003(graphics),1004(input),1007(log),1009(mount),
1011(adb),1015(sdcard_rw),1028(sdcard_r),3001(net_bt_admin),
3002(net_bt),3003/inet),3006(net_bw_stats)
```

```
private static final void enforceSystemOrRoot(String message) {
    final int uid = Binder.getCallingUid();
    if (uid != Process.SYSTEM_UID && uid != 0) {
        throw new SecurityException(message);
    }
}
```



The image shows three sequential screenshots of an Android application's permission request interface.

Screenshot 1 (Left): The title is "App permissions". It lists the permissions required by "Battery Doctor (Battery Saver)".

- Storage:** Modify/delete SD card contents
- System tools:** Bluetooth administration, change Wi-Fi state, change network connectivity, disable keylock, display system-level alerts, modify global system settings, prevent phone from sleeping, retrieve running applications, write sync settings
- Phone calls:** Read phone state and identity
- Network communication:** Create Bluetooth connections, full Internet access
- Your personal information:** Read sensitive log data

A large green "ACCEPT" button is at the bottom.

Screenshot 2 (Middle): A modal dialog box is displayed, listing the requested permissions:

- Wi-Fi state, work in settings, sync
- Internet

A large green "ACCEPT" button is at the bottom.

Screenshot 3 (Right): Another modal dialog box is displayed, listing the requested permissions:

- Wi-Fi state, modify global from sleeping, sync
- fine (GPS)
- Internet

A large green "ACCEPT" button is at the bottom.



Android permissions

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.aditya.something"
    android:versionCode="1"
    android:versionName="1.0" >
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="17" />
    ...

```

declarative security

<http://developer.android.com/guide/topics/manifest/manifest-element.html>



Android permissions

/system/etc/permissions/platform.xml

```
<permissions>
    <permission name="android.permission.INTERNET">
        <group gid="inet" />
    </permission>
    <permission name="android.permission.BLUETOOTH">
        <group gid="net_bt" />
    </permission>
    <permission name="android.permission.CAMERA">
        <group gid="camera" />
    </permission>
    ...

```

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.bar.foo"
    <permission-group
        android:name="com.bar.foo.permission-group.TEST_GROUP" ①
        android:label="@string/foo_permission_group_label"
        android:description="@string/foo_permission_group_desc" />
    <permission
        android:name="com.bar.foo.permission.PERMISSION1" ②
        android:label="@string/permission1_label"
        android:description="@string/permission1_desc"
        android:permissionGroup="com.bar.foo.permission-group.TEST_GROUP"
        android:protectionLevel="signature" />
    <permission-tree
        android:name="com.bar.foo.permission"
        android:label="@string/foo_permission_tree_label" />
    ...

```

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.bar.foo"

PackageManager pm = getPackageManager();
PermissionInfo permission = new PermissionInfo();
permission.name = "com.bar.foo.permission.PERMISSION2";
permission.labelRes = R.string.permission2_label;
permission.protectionLevel = PermissionInfo.PROTECTION_SIGNATURE;
boolean added = pm.addPermission(permission);
Log.d(TAG, "permission added: " + added);

        android:description="@string/permission1_desc"
        android:permissionGroup="com.bar.foo.permission-group.TEST_GROUP"
        android:protectionLevel="signature" />

<permission-tree
    android:name="com.bar.foo.permission"
    android:label="@string/foo_permission_tree_label" />
...

```

①

②

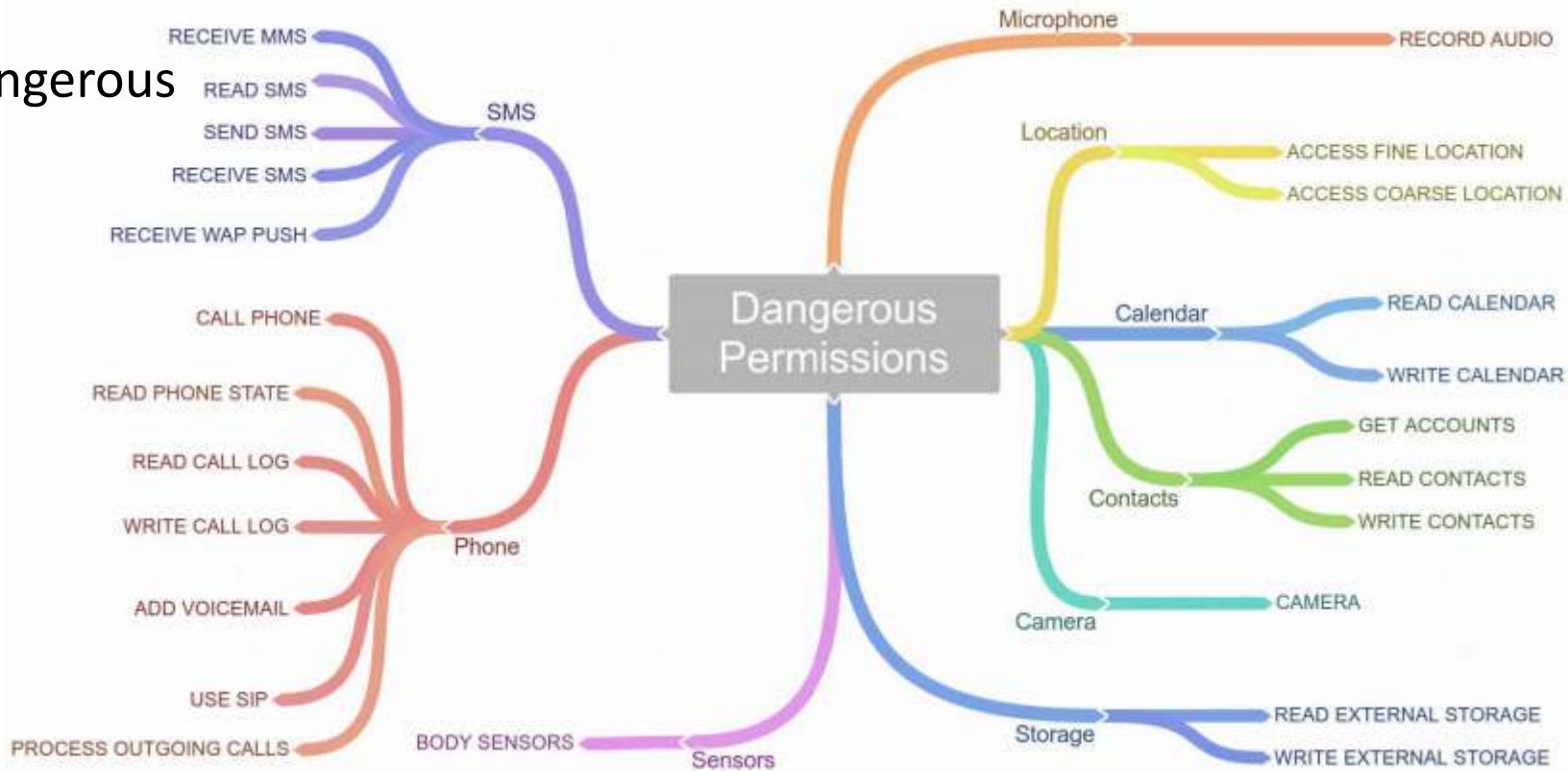
③

Android permissions



Permission Protection Levels:

- normal
- dangerous





Android permissions

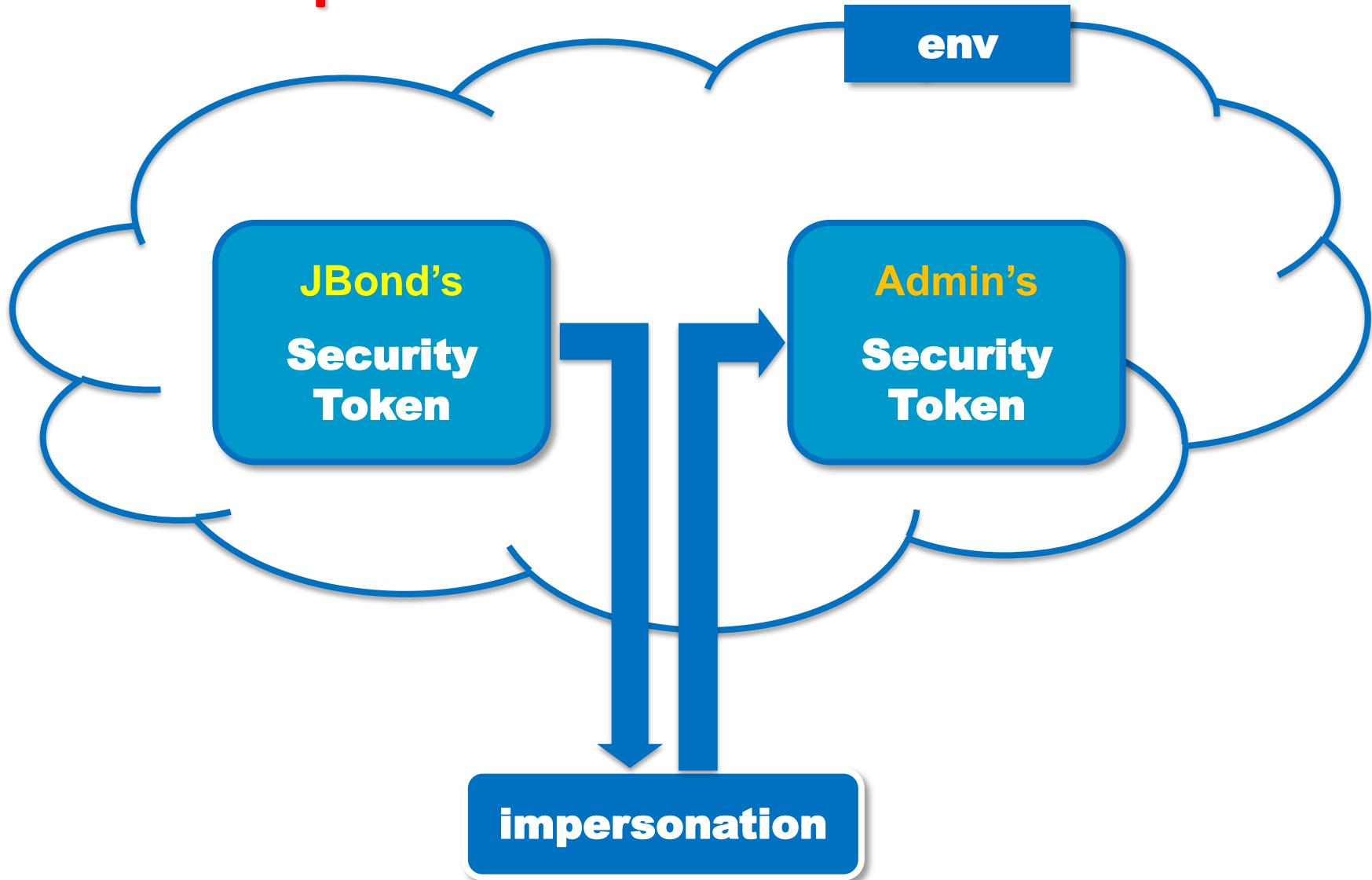
Permission Protection Levels:

- ➔ normal
- ➔ dangerous (e.g. READ_SMS, CAMERA)
- ➔ signature (only applications that are signed with the same key)
- ➔ system (only system image applications or /system)
- ➔ signatureOrSystem (/system/priv-app/)
- ➔ development (only privileged context ADB)
- ➔ *applications that do not have any intention of sharing data or functionality with applications from other developers should always define permissions with the signature protection level*

Impersonation



Impersonation



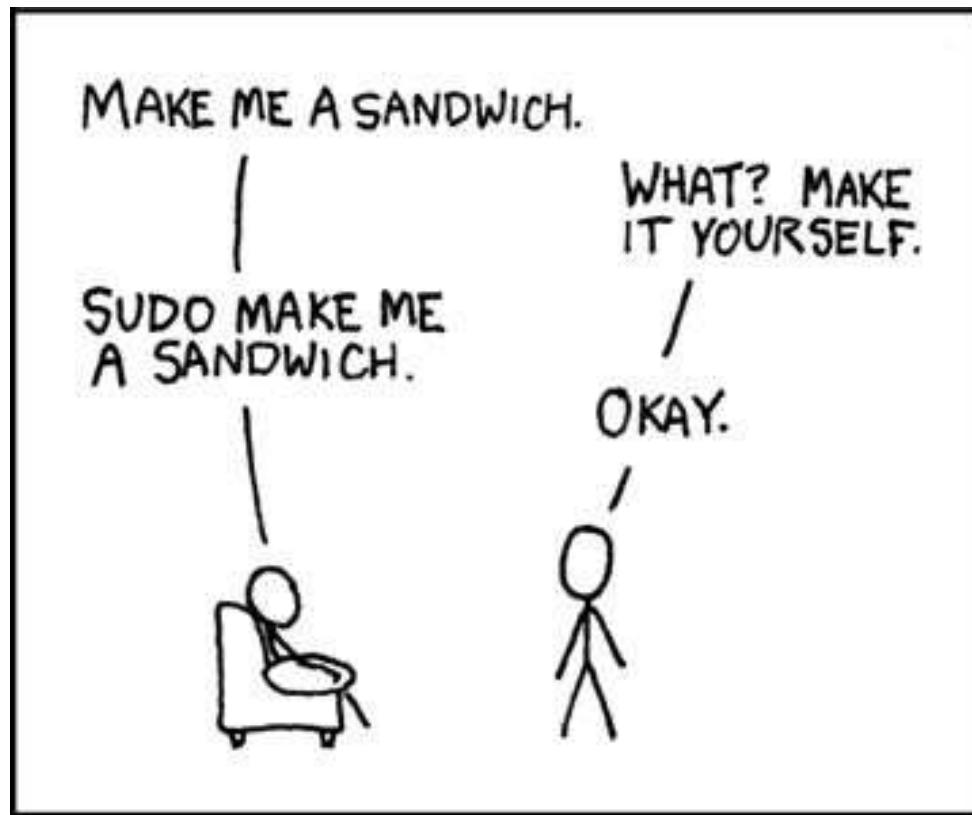
Impersonation

Unix: su

sudo

SUID/SIGID

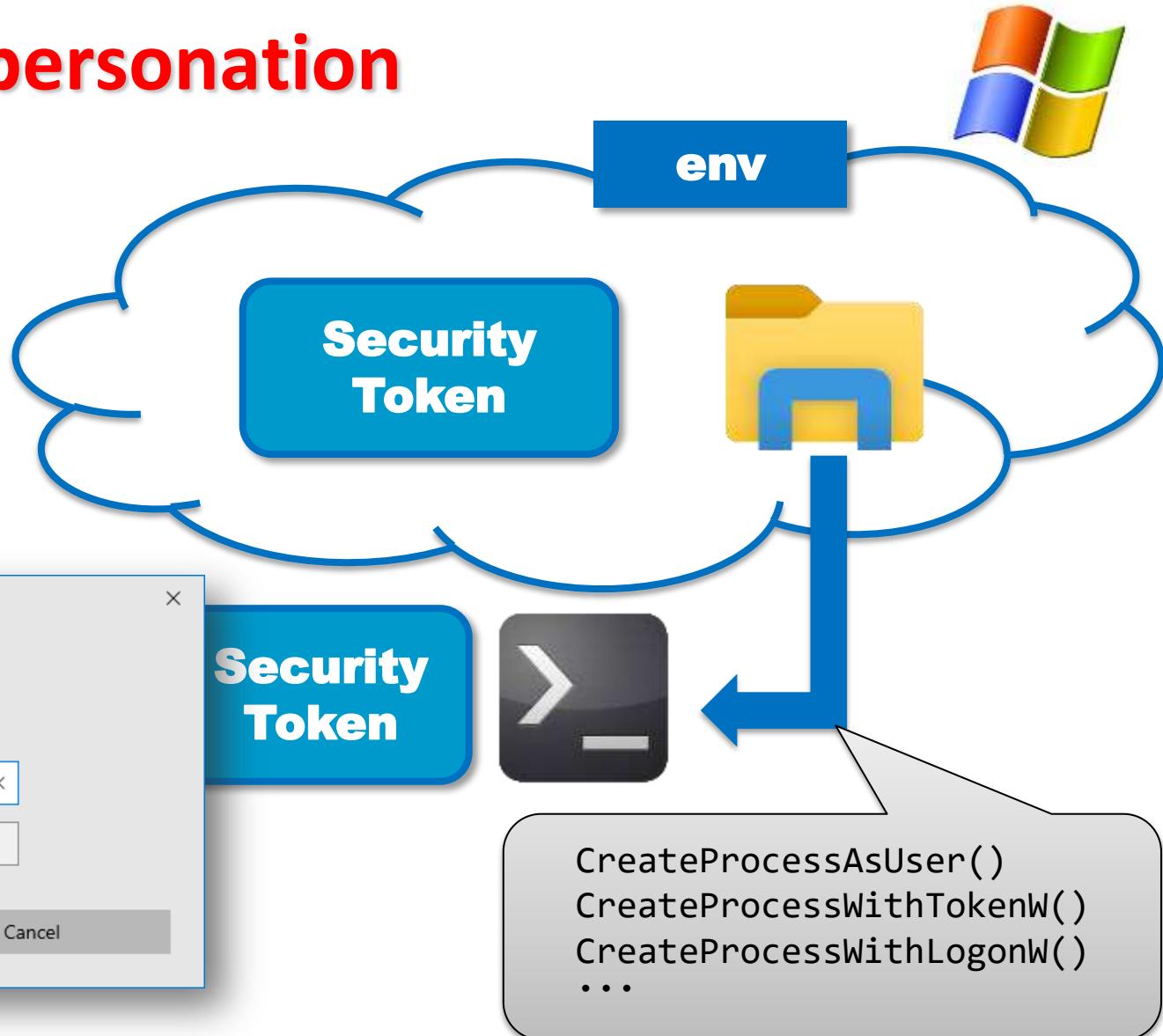
Windows: runas



(Randall Munroe) <https://xkcd.com>

Impersonation

Windows





Impersonation

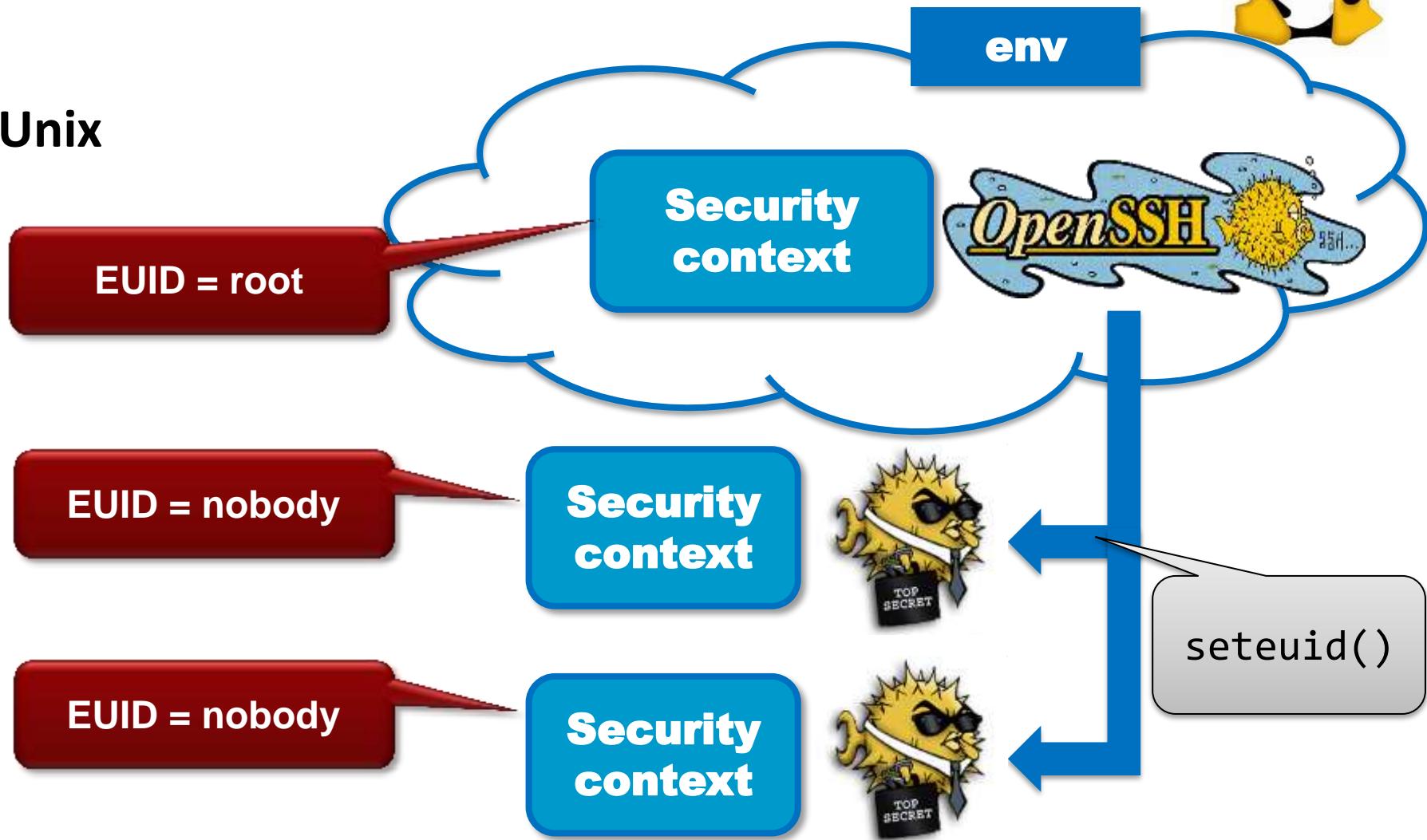
Android

- ➔ root/system users do not have a password (no /etc/passwd)
- ➔ by default no su or sudo
- ➔ /system partition is mounted as nosuid
- ➔ Android 4.3 has removed all SUID/SGID system programs and added support for → POSIX capabilities instead

Privilege separation



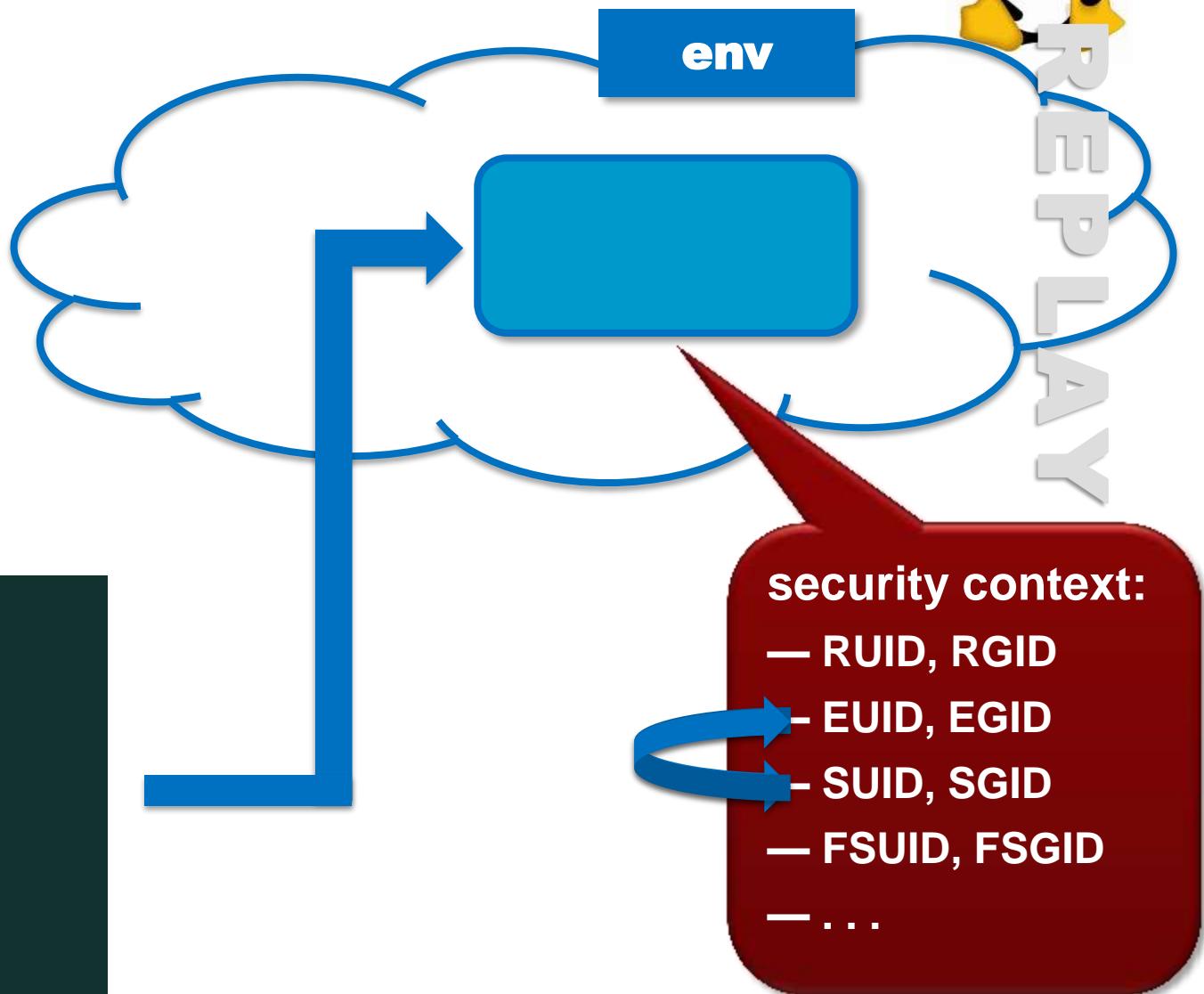
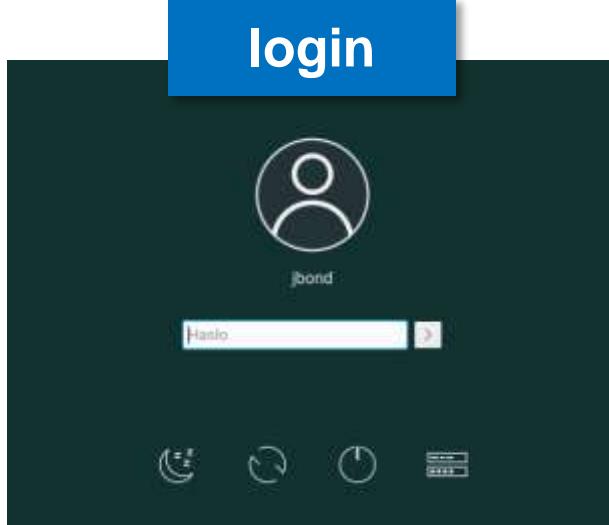
Unix



Privilege separation



Unix



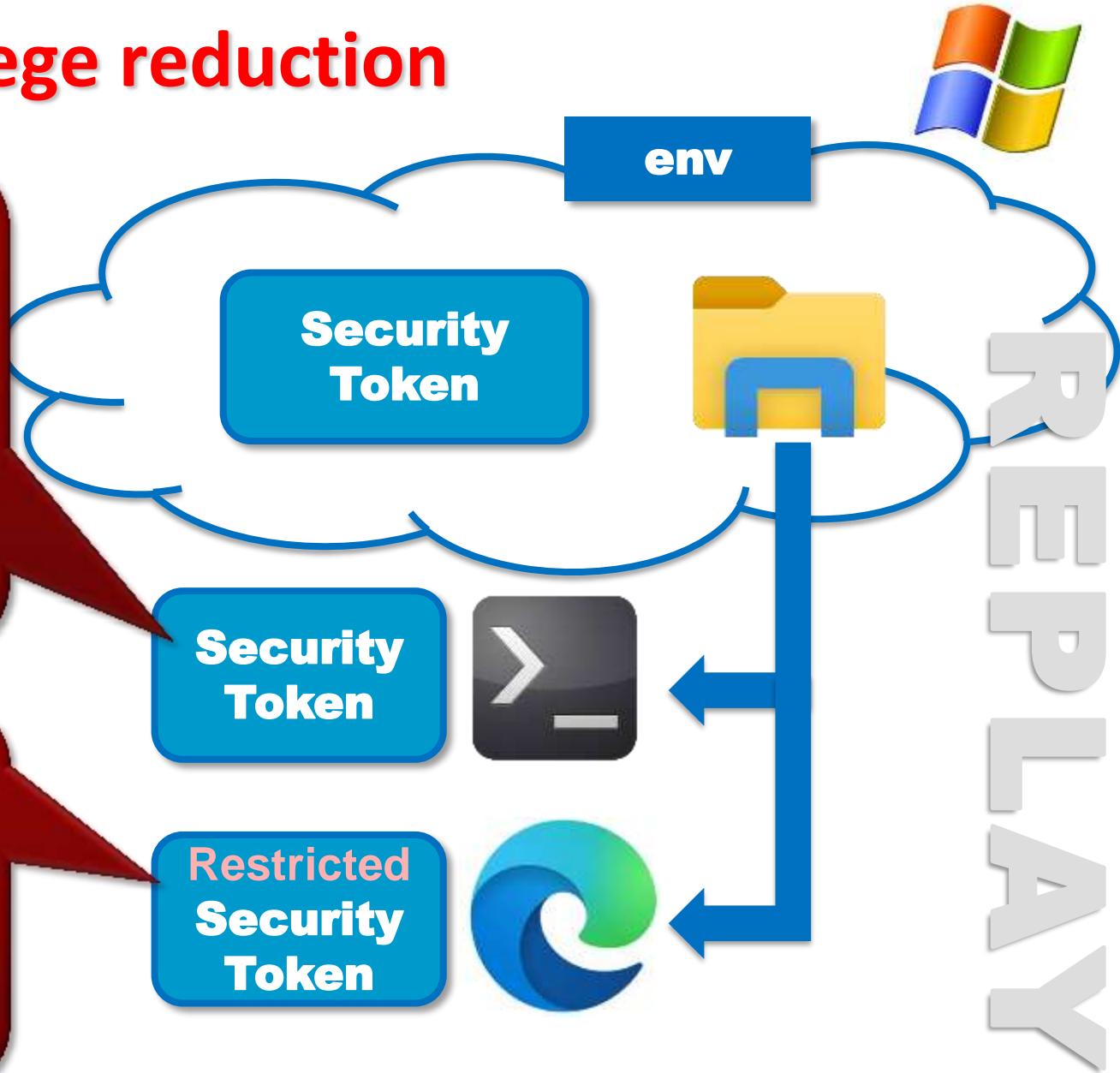
Privilege reduction

security context:

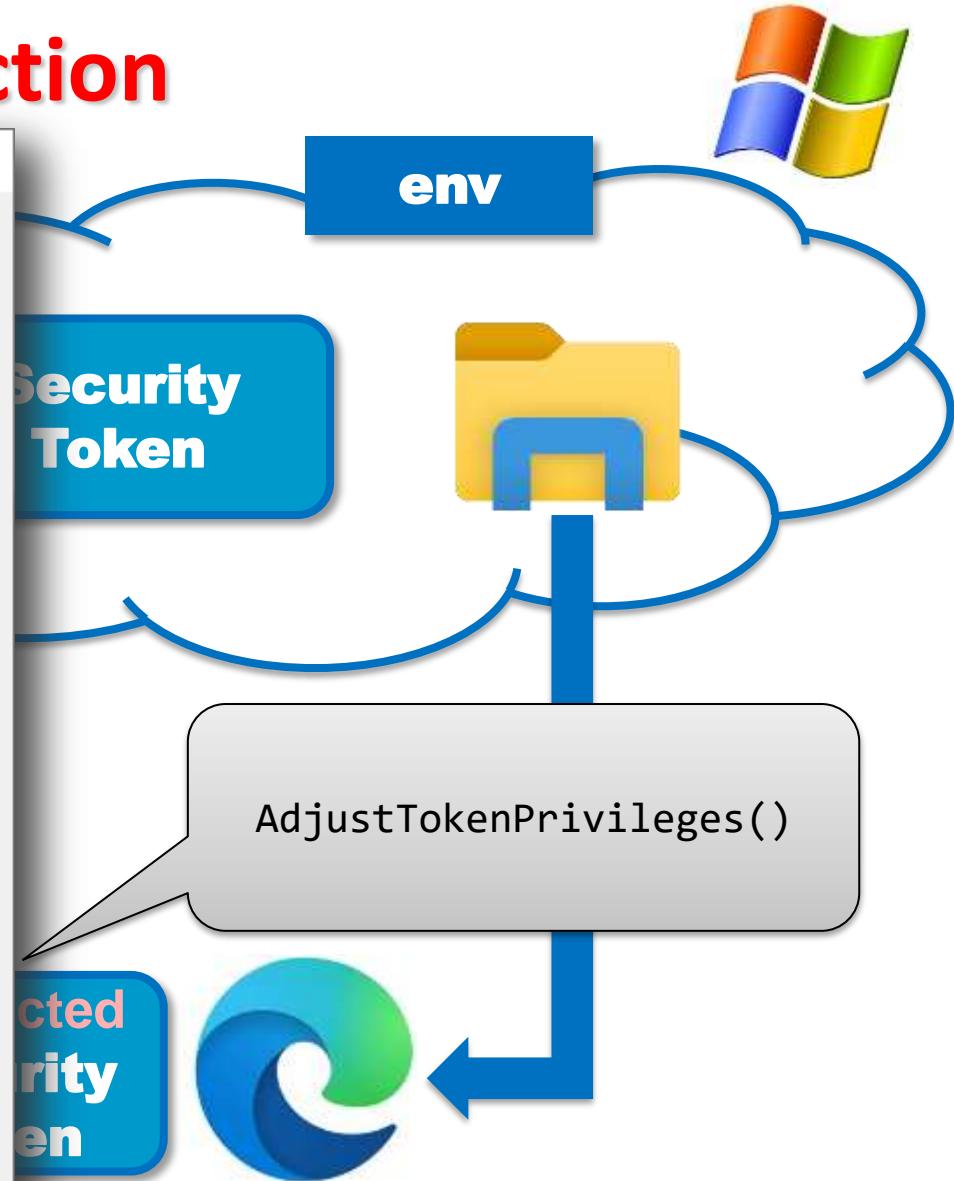
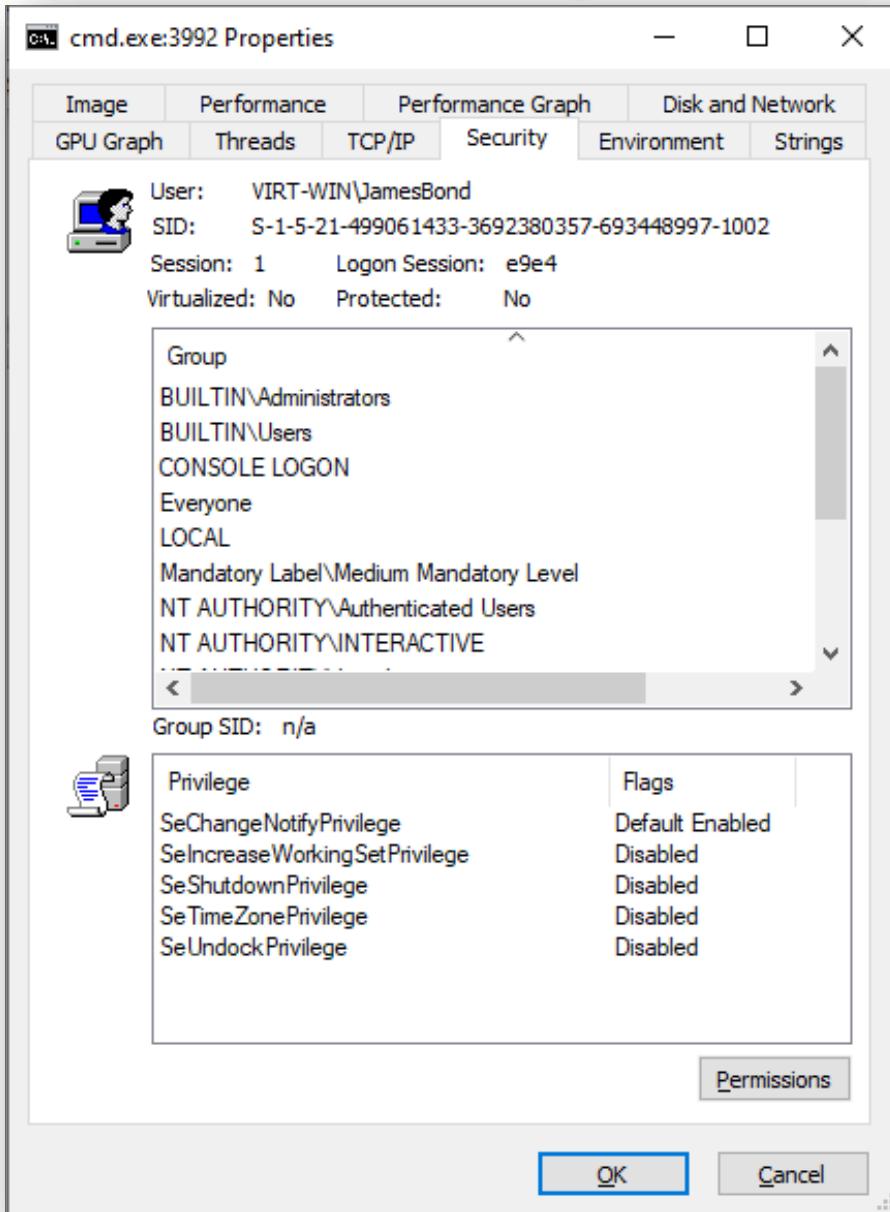
- ...
- privs:
 - 1) ...
 - 2) ...
 - 3) ...
 - 4) ...

security context:

- ...
- privs:
 - 2) ...
 - 4) ...



Privilege reduction



Privilege reduction



POSIX capabilities (CAP)

- ➔ CAP_NET_ADMIN
- ➔ CAP_NET_RAW
- ➔ CAP_KILL
- ➔ CAP_DAC_OVERRIDE
- ➔ CAP_FOWNER
- ➔ CAP_SETUID
- ➔ CAP_MKNOD
- ➔ CAP_SYS_TIME
- ➔ CAP_SYS_NICE
- ➔ CAP_MAC_ADMIN
- ➔ ...



Privilege reduction

POSIX capabilities (CAP)

- CAP_NET_ADMIN
- CAP_NET_RAW
- CAP_KILL
- CAP_DAC_OVERRIDE
- CAP_FOWNER
- CAP_SETUID
- CAP_MKNOD
- CAP_SYS_TTY_NAME
- CAP_SYS_NICE
- CAP_MAC_ADMIN
- ...

```
$ ps -fc sshd
root    2238   1  00:00:00 /usr/sbin/sshd -D
$ getpcaps 2238
Capabilities for `2238': = cap_chown,
cap_fowner, cap_fsetid, cap_kill, cap_setgid,
cap_setuid, cap_setpcap, cap_net_bind_service,
cap_net_broadcast, cap_net_admin, cap_net_raw,
...
```



Privilege reduction

POSIX capabilities (CAP)

CAP for programs

```
$ setcap cap_net_raw+ep /usr/bin/ping  
$ getcap /usr/bin/ping  
/usr/bin/ping = cap_net_raw+ep
```

CAP for users

```
cap_net_raw    jbond  
cap_sys_ptrace jdeveloper
```

/etc/security/capability.conf

+ PAM module: pam_cap

Privilege reduction



Windows privileges ≈ CAP

→ e.g. SeCreateSymbolicLinkPrivilege

The screenshot shows the Windows Local Security Policy snap-in. The left pane displays a tree structure with nodes like Security Settings, Account Policies, Local Policies (with Audit Policy selected), and various policy types. The right pane shows the details of the selected 'Audit Policy' node, specifically the 'User Rights Assignment' tab. A list of user rights is shown, with 'Create symbolic links' being highlighted.

Privilege	Flags
SeBackupPrivilege	Disabled
SeChangeNotifyPrivilege	Default Enabled
SeCreateGlobalPrivilege	Default Enabled
SeCreatePagefilePrivilege	Disabled
SeCreateSymbolicLink Privilege	Disabled
SeDebugPrivilege	Disabled
SeDelegateSessionUserImpersonatePrivilege	Disabled
SeInheritablePrivilege	Default Enabled

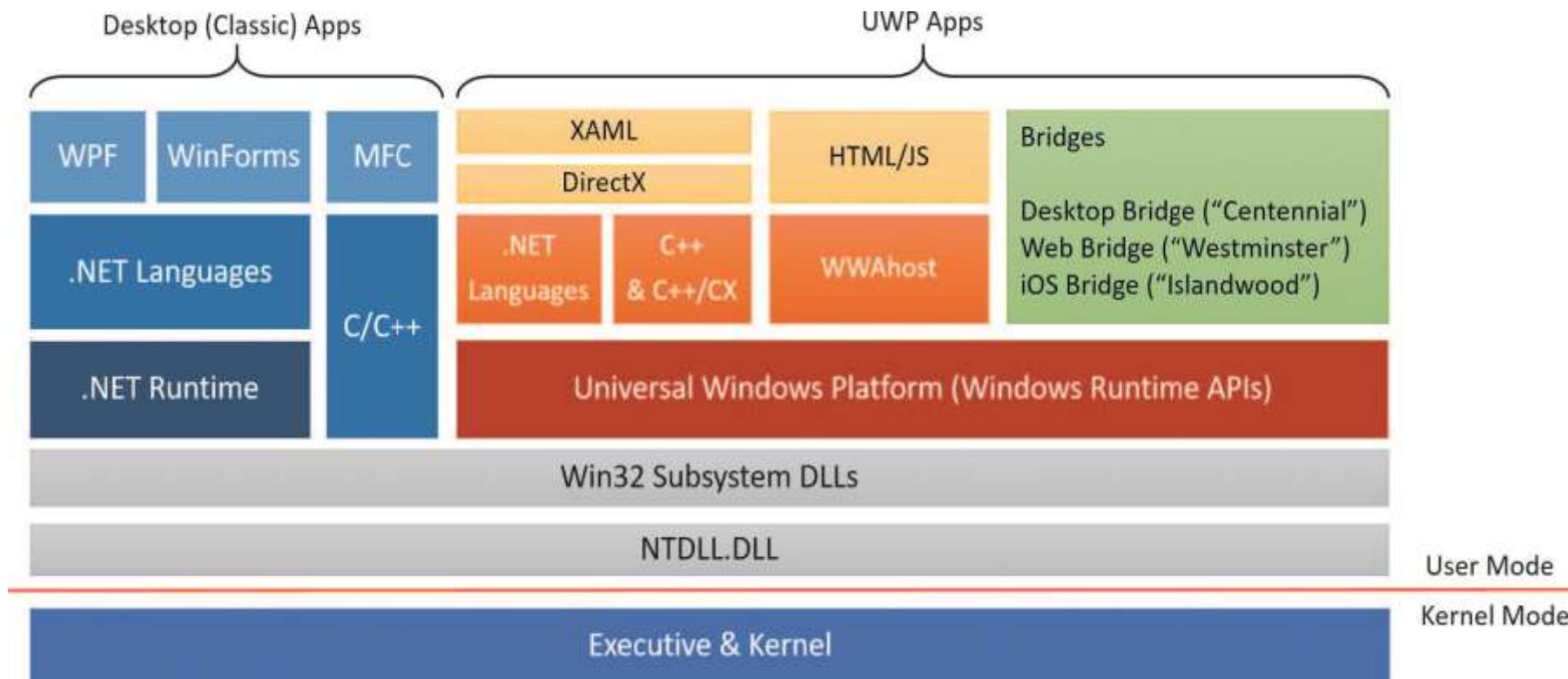
The screenshot shows the Windows Task Manager displaying the properties of the 'cmd.exe:3992' process. The 'Security' tab is selected. It shows the user 'VIRT-WIN\JamesBond' and their SID. The 'Group' section lists several security groups, and the 'Privilege' section shows the current state of various Windows privileges. The 'SeCreateSymbolicLink Privilege' is explicitly listed as disabled.

Privilege	Flags
SeBackupPrivilege	Disabled
SeChangeNotifyPrivilege	Default Enabled
SeCreateGlobalPrivilege	Default Enabled
SeCreatePagefilePrivilege	Disabled
SeCreateSymbolicLink Privilege	Disabled
SeDebugPrivilege	Disabled
SeDelegateSessionUserImpersonatePrivilege	Disabled
SeInheritablePrivilege	Default Enabled

Privilege reduction



Universal Windows Platform capabilities ≈ Android permissions



Privilege reduction



Universal Windows Platform capabilities ≈ Android permissions

Win8MetroTest - Microsoft Visual Studio

FILE EDIT VIEW PROJECT BUILD DEBUG TEAM SQL TOOLS TEST ARCHITECTURE ANALYZE WINDOW HELP

Local Machine Debug Any CPU

Package.appxmanifest

The properties of the deployment package for your app are contained in the app manifest file. You can use the Manifest Designer to set or modify one or more of the properties.

Application UI Capabilities Declarations Packaging

Use this page to specify system features or devices that your app can use.

Capabilities:

- Documents Library
- Enterprise Authentication
- Internet (Client)
- Internet (Client & Server)
- Location
- Microphone
- Music Library
- Pictures Library
- Private Networks (Client & Server)
- Proximity
- Removable Storage
- Shared User Certificates
- Videos Library
- Webcam

Description:

Provides inbound and outbound access to Intranet networks that have an authenticated domain controller, or that the user has designated as either home or work networks. Inbound access to critical ports is always blocked.

[More information](#)

Micha

Authorization & Access Control



1. DAC ACL examples:

- ➔ POSIX ACL
- ➔ Windows DACL

2. Impersonation (→ authentication)

3. Privilege separation

= change of security context through forking a child process

4. Privilege reduction

- ➔ POSIX CAP
- ➔ Windows Restricted Token

Guidelines & Recommendations

<https://www.cisecurity.org/cis-benchmarks/>



CIS Hardened Images Support CIS WorkBench Sign-in



CIS Benchmarks™



With our global community of cybersecurity experts, we've developed CIS Benchmarks: more than 100 configuration guidelines across 25+ vendor product families to safeguard systems against today's evolving cyber threats.

[Join a Community](#)



Feedback

[Operating Systems](#)

[Server Software](#)

[Cloud Providers](#)

[Mobile Devices](#)

[Network Devices](#)

[Desktop Software](#)

[Multi Function Prin...](#)

[Cloud Providers](#)

Alibaba Cloud

Expand to see related content

[Download CIS Benchmark](#)

[Operating Systems](#)

Amazon Linux

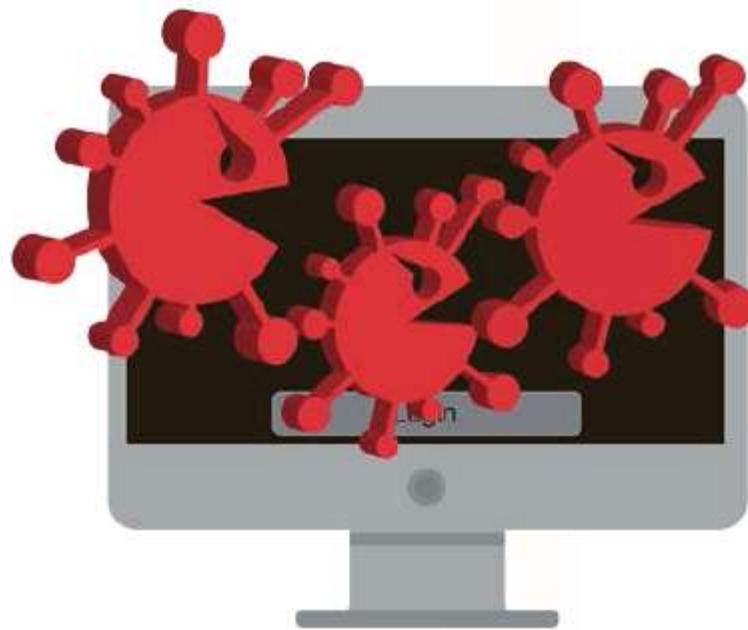
Expand to see related content

[Download CIS Benchmark](#)

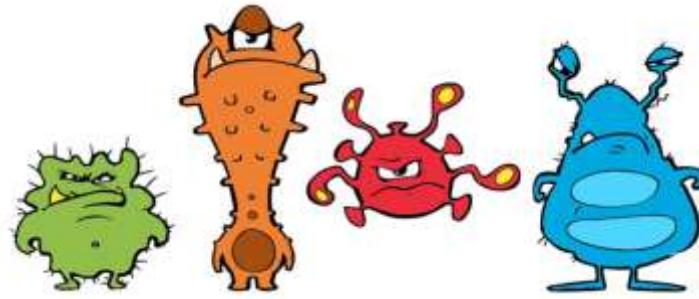
CIS Hardened Image and Build Kit also available

[Linux](#)

MALWARE



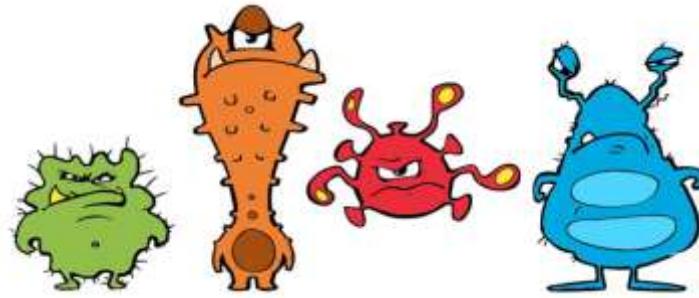
Malware



- ▶ viruses
- ▶ network worms
 - 💣 Internet Worm (© Robert Morris junior, 1988)
- ▶ Trojan horses

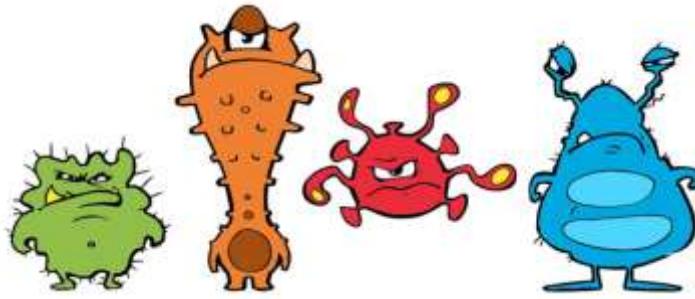


Malware



- ➔ viruses
- ➔ network worms
 - 💣 Internet Worm (© Robert Morris junior, 1988)
- ➔ Trojan horses
- ➔ rootkits
 - ➔ kernel rootkits
 - ➔ BIOS/UEFI rootkits (ACPI = Advanced Configuration & Power Interface)
 - ➔ VM rootkits (e.g. Blue Pill)
 - ➔ hypervisor rootkits

Malware



There's a new virus on the loose that's worse than anything we've seen before! It gets in through the power line. It works by changing the serial port pinouts, and by reversing the direction one's disks spin.

Don't use the powerline. Don't use batteries either, since there are rumours that this virus has invaded most major battery plants, and is infecting the positive poles of the batteries. (You might try hooking up just the negative pole.) Don't use Internet. Don't read messages. No, not even this one!

<https://www.hoax-slayer.net>

HOAX

Malware



Ransomware

Wana Decrypt0r 2.0

English

Ooops, your files have been encrypted!

What Happened to My Computer?
Your important files are encrypted.
Many of your documents, photos, videos, databases and other files are no longer accessible because they have been encrypted. Maybe you are busy looking for a way to recover your files, but do not waste your time. Nobody can recover your files without our decryption service.

Can I Recover My Files?
Sure. We guarantee that you can recover all your files safely and easily. But you have not so enough time.
You can decrypt some of your files for free. Try now by clicking <Decrypt>. But if you want to decrypt all your files, you need to pay.
You only have 3 days to submit the payment. After that the price will be doubled. Also, if you don't pay in 7 days, you won't be able to recover your files forever.
We will have free events for users who are so poor that they couldn't pay in 6 months.

How Do I Pay?
Payment is accepted in Bitcoin only. For more information, click <About bitcoin>. Please check the current price of Bitcoin and buy some bitcoins. For more information, click <How to buy bitcoins>. And send the correct amount to the address specified in this window. After your payment, click <Check Payment>. Best time to check: 9:00am - 11:00am
GMT +00:00, Monday, April 24, 2017

Send \$300 worth of bitcoin to this address:

bitcoin ACCEPTED HERE

12t9YDPgwueZ9NyMgw519p7AA8isjr6SMw

Copy

[About bitcoin](#)

[How to buy bitcoins?](#)

[Contact Us](#)

[Check Payment](#)

[Decrypt](#)

Payment will be raised on
5/16/2017 00:47:55

Time Left
02:23:57:37

Your files will be lost on
5/20/2017 00:47:55

Time Left
06:23:57:37

Malware?



Greyware (spyware, trackware)

Task Scheduler

File Action View Help

Application Experience ApplicationData AppxDeploymentClient Autochk BitLocker Bluetooth BrokerInfrastructure CertificateServicesClient Chkdsk Clip CloudExperienceHost Customer Experience Improvement Data Integrity Scan Defrag Device Information Device Setup DeviceDirectoryClient Diagnosis

Name	Status	Triggers	Next Run Time	Last Run Time	Last Run Result
DmClient	Ready		11/30/1999 12:00:00 AM	11/30/1999 12:00:00 AM	The task has not yet run.
DmClientOn...	Ready	Custom Trigger	1/11/2021 10:15:00 AM	1/11/2021 10:15:00 AM	The operation completed successfully.

General Triggers Actions Conditions Settings History (disabled)

Name: DmClient
Location: \Microsoft\Windows\Feedback\Siuf
Author: Microsoft Windows Feedback
Description: Update SIUF strings

Malware



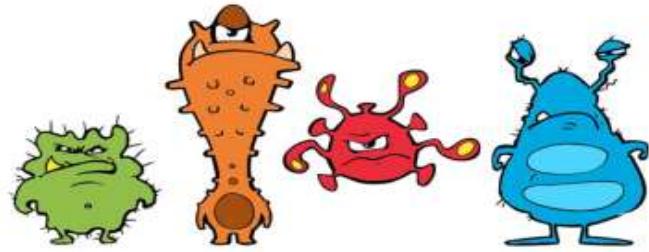
The best antivirus software for Android

July 2017

	Name	AV-TEST	Protection	Usability	
AhnLab	AhnLab V3 Mobile Security 3.1		●●●●●	●●●●●	►
Antiy	Antiy AVL 2.5		●●●●●	●●●●●	►
avast	Avast Mobile Security 6.1		●●●●●	●●●●●	►
Bitdefender	Bitdefender Mobile Security 3.2		●●●●●	●●●●●	►
G DATA	G Data Internet Security 26.0		●●●●●	●●●●●	►
IKARUS	Ikarus mobile.security 1.7		●●●●●	●●●●●	►
KASPERSKY	Kaspersky Lab Internet Security 11.13		●●●●●	●●●●●	►
McAfee	McAfee Mobile Security 4.9		●●●●●	●●●●●	►
PSafe	PSafe DFNDR 4.0		●●●●●	●●●●●	►
SOPHOS	Sophos Mobile Security 7.0		●●●●●	●●●●●	►

<https://www.av-test.org>

Malware



blackhat[®]
USA 2017

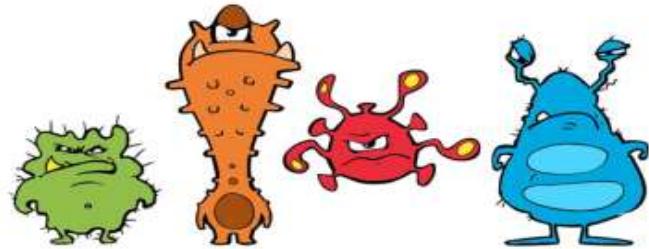
JULY 22-27, 2017
MANDALAY BAY / LAS VEGAS

AVPASS: Automatically Bypassing Android Malware Detection System

Jinho Jung, Chanil Jeon, Max Wolotsky, Insu Yun, and Taesoo Kim
Georgia Institute of Technology, July 27, 2017

#BHUSA / @BLACKHATEVENTS

Malware



black hat
USA 2017

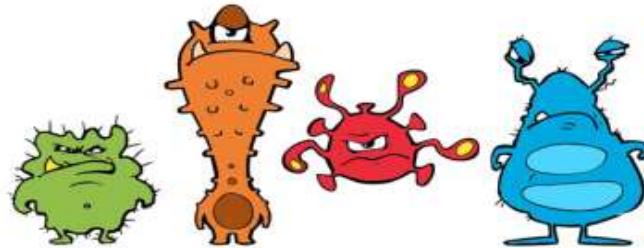
Evaluation: Bypassing AVs

- General bypass ability

Category	Avg. Detections	Detection Ratio
Average Detections	38 / 58	65%

* Experiment in July / 2017, Test with 2,000 malware

Malware



black hat
USA 2017

Evaluation: Bypassing AVs

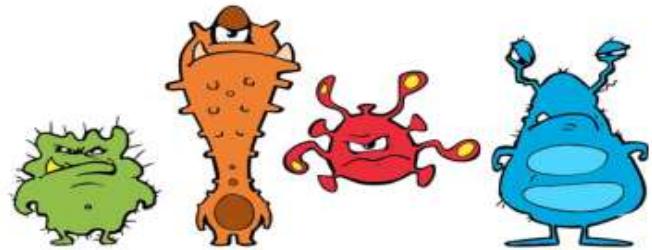
- General bypass ability

Category	Avg. Detections	Detection Ratio
Average Detections	38 / 58	65%
After AVPASS	3.42 / 58	5.8%

* Experiment in July / 2017, Test with 2,000 malware



Malware



EPP = End-Point Protection = Antivirus

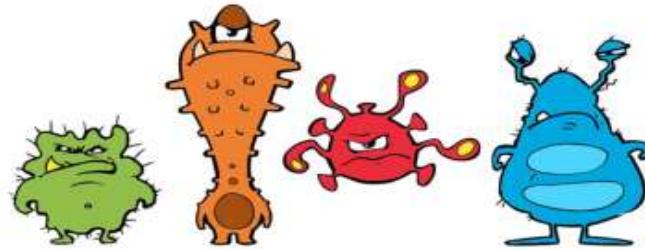
EDR = Endpoint Detection and Response

XDR = Extended Detection and Response

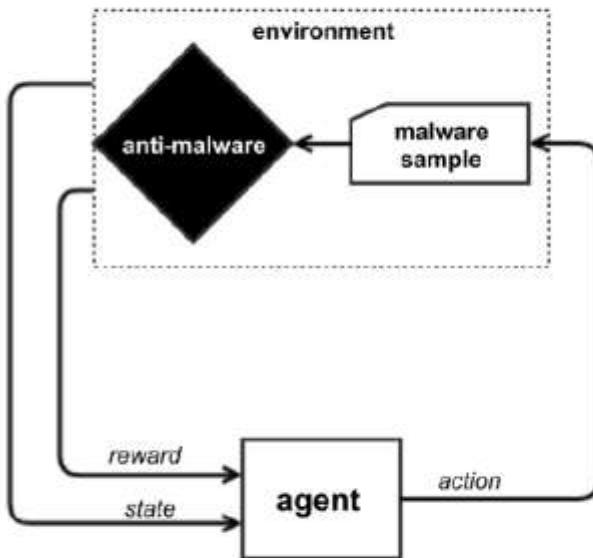
MDR = Managed Detection and Response



Malware



Bot vs. Bot: Evading Machine Learning Malware Detection.
Hyrum Anderson. Blackhat USA 2017.



• Environment

- A malware sample (*Windows PE*)
- Buffet of malware mutations
 - *preserve format & functionality*
- Reward from static malware classifier

• Agent

- Input: **environment state** (*malware bytes*)
- Output: **action** (*stochastic*)
- Feedback: **reward** (*AV reports benign*)

Features

- Static Windows PE file features compressed to 2350 dimensions
 - General file information (size)
 - Header info
 - Section characteristics
 - Imported/exported functions
 - Strings
 - File byte and entropy histograms

- Feed a neural network to choose the best action for the given "state"

Functionality-preserving mutations:

• Create

- New Entry Point (w/ trampoline)
- New Sections

• Add

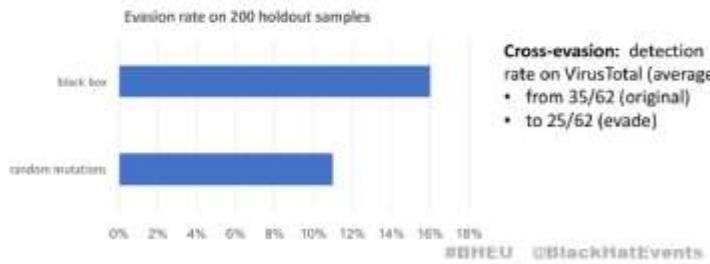
- Random Imports
- Random bytes to PE overlay
- Bytes to end of section

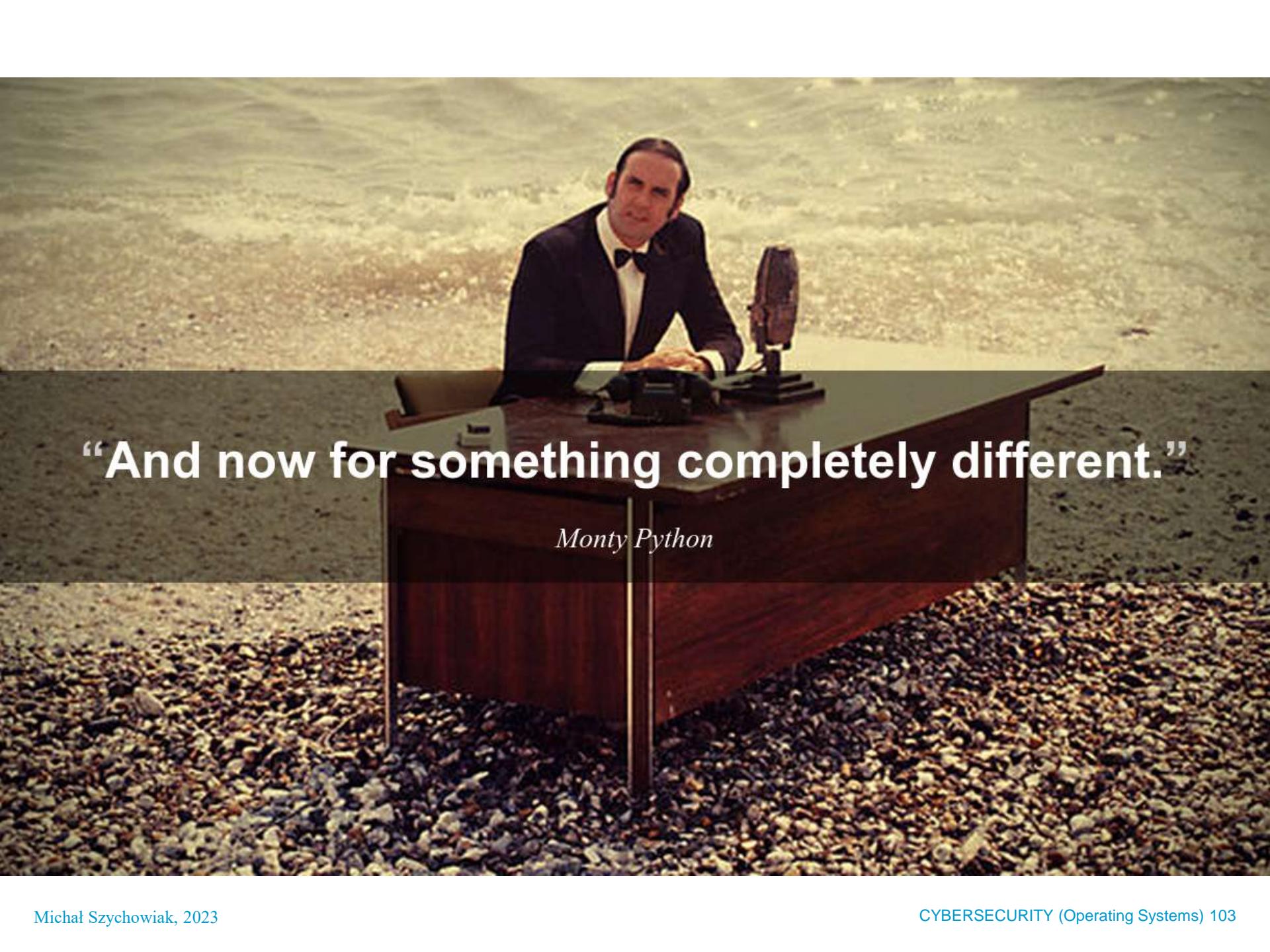
• Modify

- Random sections to common name
- (break) signature
- Debug info
- UPX pack / unpack
- Header checksum
- Signature

Evasion Results

- Agent training: 15 hours for 100K trials (~10K games x 10 turns ea.)
- Using malware samples from VirusShare



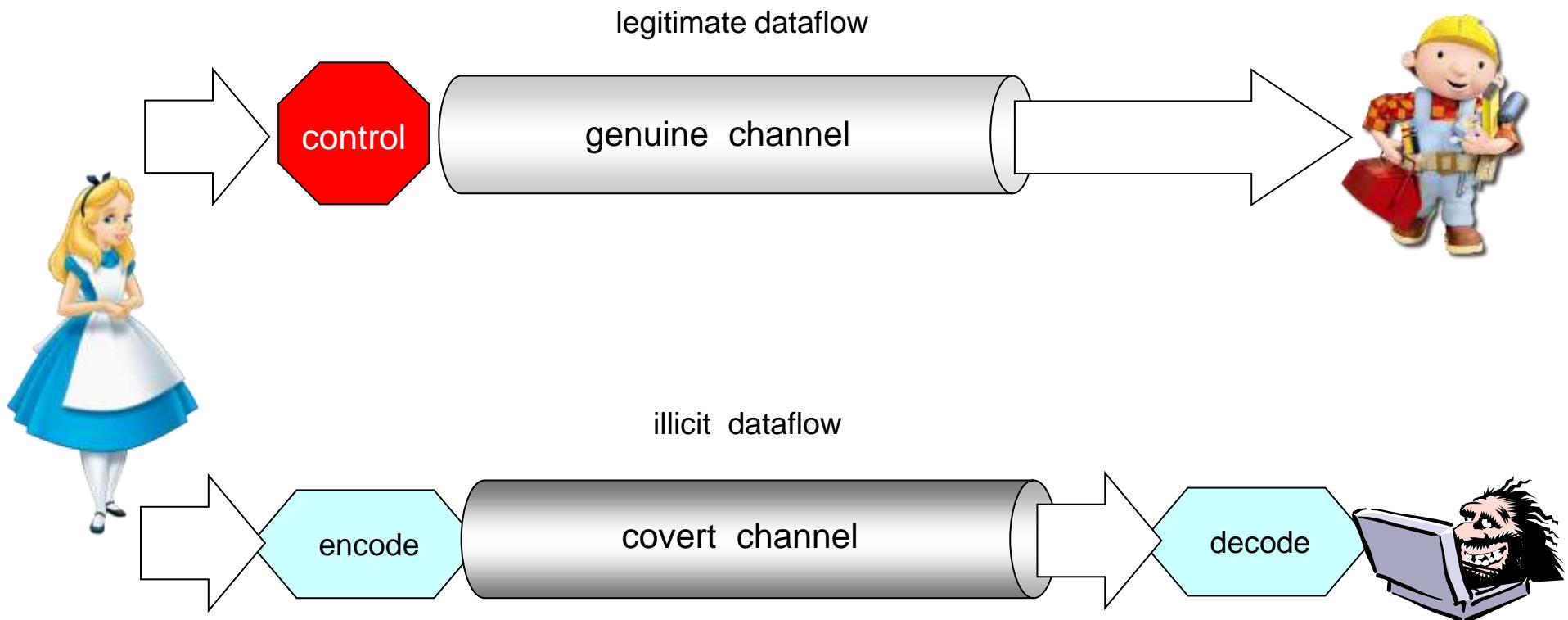
A still from the opening sequence of Monty Python's Flying Circus. Terry Gilliam is seated at a large, dark wooden desk on a beach. He is wearing a dark suit, white shirt, and bow tie. He is looking directly at the camera with a serious expression. On the desk in front of him is a vintage-style microphone on a stand. The background shows the ocean waves crashing onto a sandy beach.

“And now for something completely different.”

Monty Python

COVERT CHANNELS

Covert Channels



Covert Channels

Sample covered channels:

- timing-based
- print spooler
- filesystem
- DNS
- IP TTL / Hop Limit

Sample defense:

- communication noise (for the timing-based channels):
 $1 \text{ MB} \cdot 1\text{b} / 10\text{s} = 2 \frac{1}{2} \text{ year}$



“That's all folks!”