No Universal Default Passwords

Michael Richardson,

Sandelman Software Works

Emily Taylor,

Oxford Information Labs

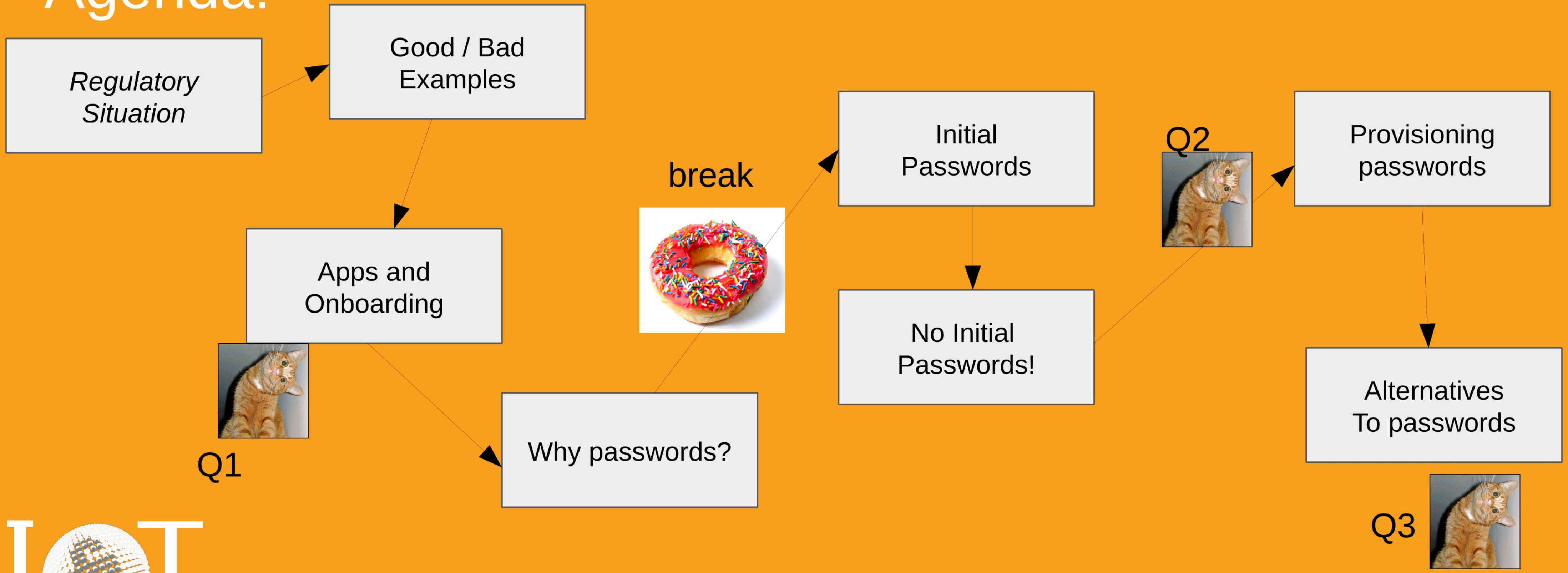
Security and Technical Education Programme (STEP)

IoT Security Foundation

OXFORD INFORMATION LABS

Password Course Overview

Agenda!



Regulatory Situation

Good / Bad Examples

Apps and Onboarding

Q1

Why passwords?

break

Initial Passwords

No Initial Passwords!

Q2

Provisioning passwords

Alternatives To passwords

Q3

IoT Security Foundation

OXFORD INFORMATION LABS

Examples of Bad Password Uses

The most obvious problem is:

```
   Login: admin
Password: admin
```

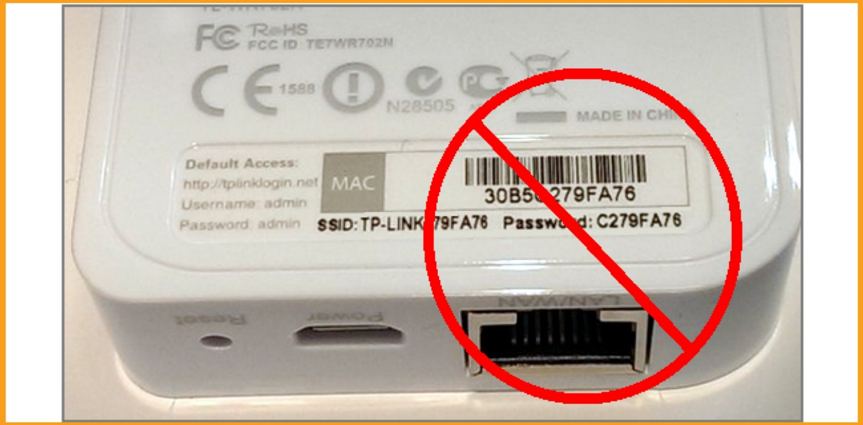
This would not be too horrible, if the password was always changed.

But, they are not getting changed.
943 devices use this, according to
https://bestvpn.org/default-router-passwords/

This device uses it’s ethernet address as the password:

```
   Login: admin
Password: C279FA76
```

TP-Link has been assigned 134 IEEE blocks of OUI (ethernet addresses). An exhaustive search is not that hard, and malware on a PC can trivially find out the MAC address without any search.

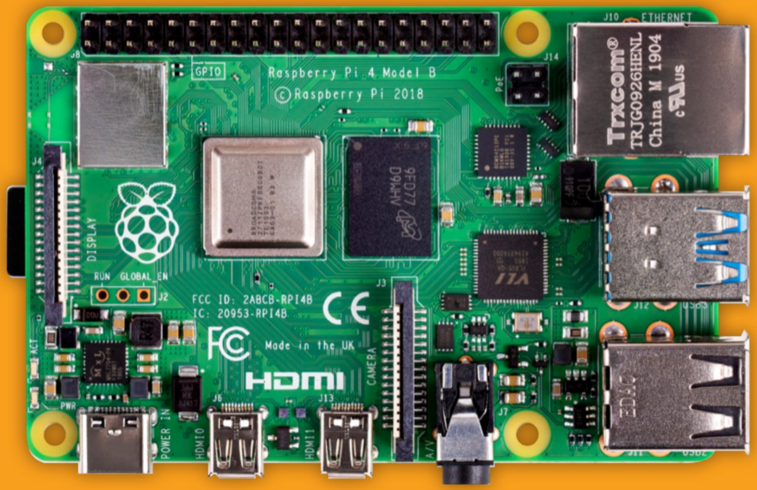


More Bad initial Passwords

Many devices use Raspberry PIs and Raspbian

```
   Login: pi
Password: raspberry
```

(50% of RPIs are going into industrial uses)

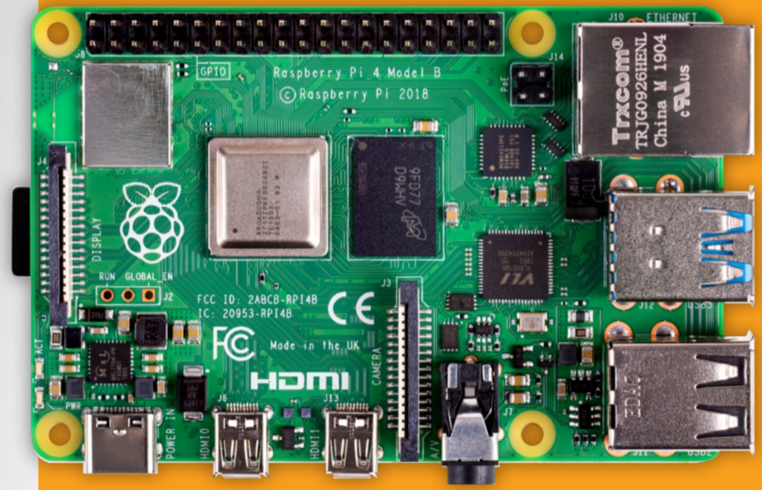


More Bad initial Passwords

Many devices use Raspberry PIs and Raspbian

```
   Login: pi
Password: raspberry
```

(50% of RPIs are going into industrial uses)

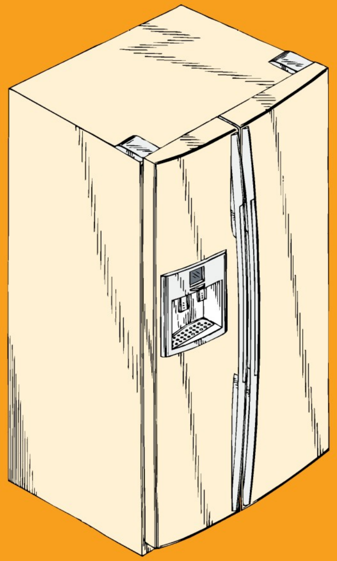


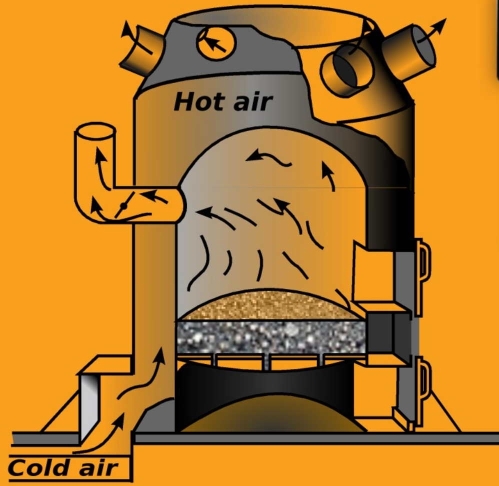
More Bad initial Passwords

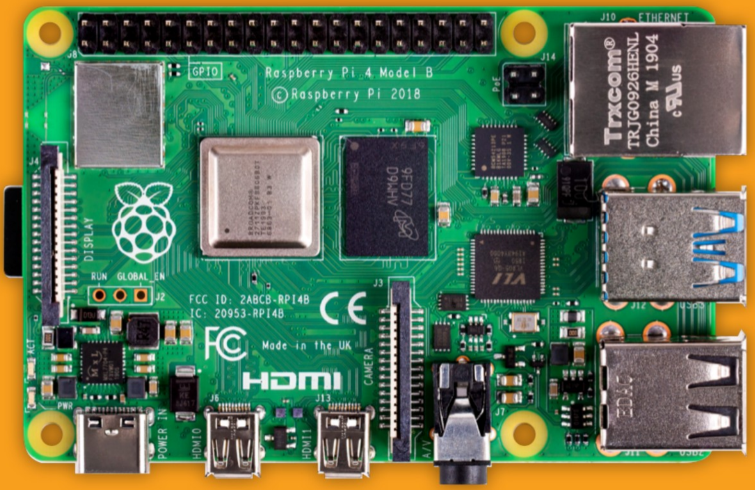
Many devices use Raspberry PIs and Raspbian

Login: pi
Password: raspberry

(50% of RPIs are going into industrial uses)





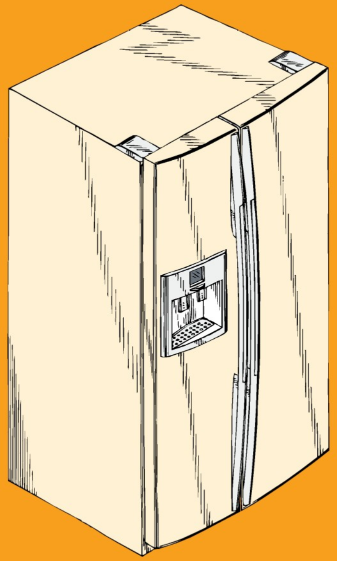


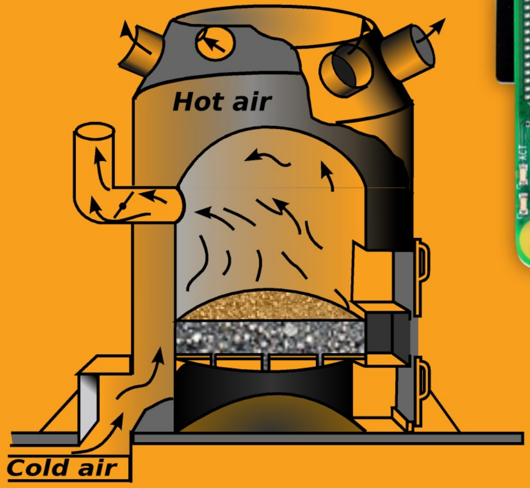
More Bad initial Passwords

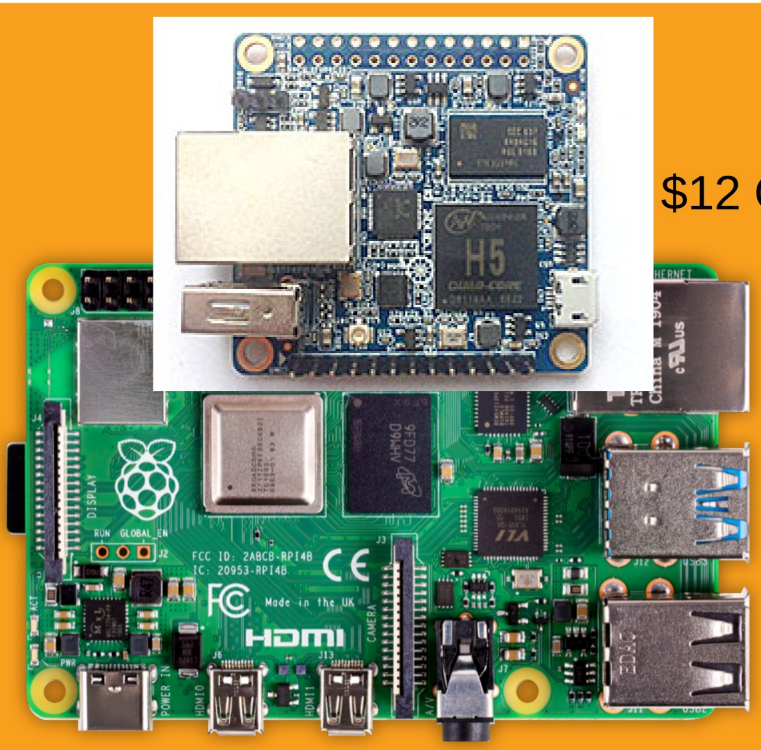
Many devices use Raspberry PIs and Raspbian

```
   Login: pi
Password: raspberry
```

(50% of RPIs are going into industrial uses)







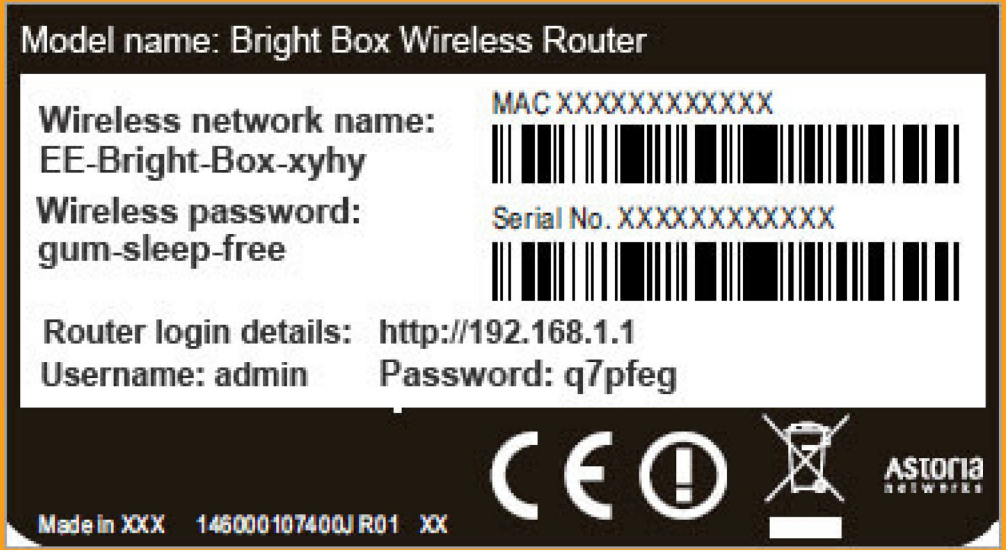
$12 QTY 1

Some good password stories

This one is a good example:

Login: admin
Password: q7pfeg

The password looks random.



Model name: Bright Box Wireless Router

Wireless network name:
EE-Bright-Box-xyhy

Wireless password:
gum-sleep-free

MAC XXXXXXXXXXXX

Serial No. XXXXXXXXXXXX

Router login details: http://192.168.1.1
Username: admin
Password: q7pfeg

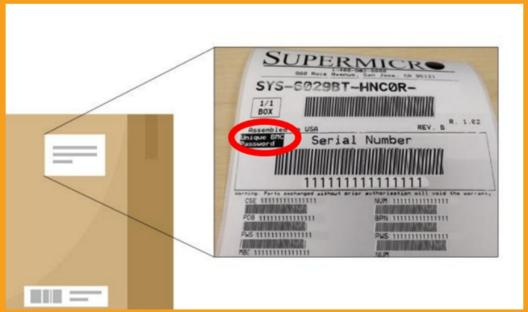
Made in XXX 146000107400J R01 XX

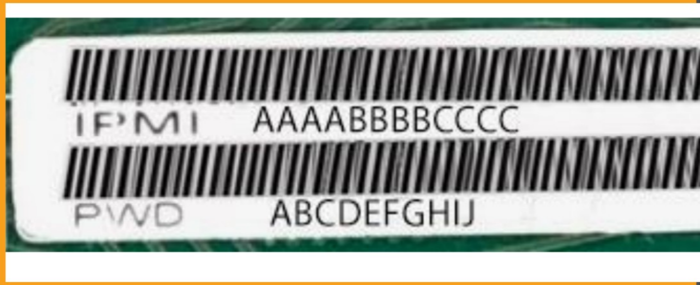
Astoria networks

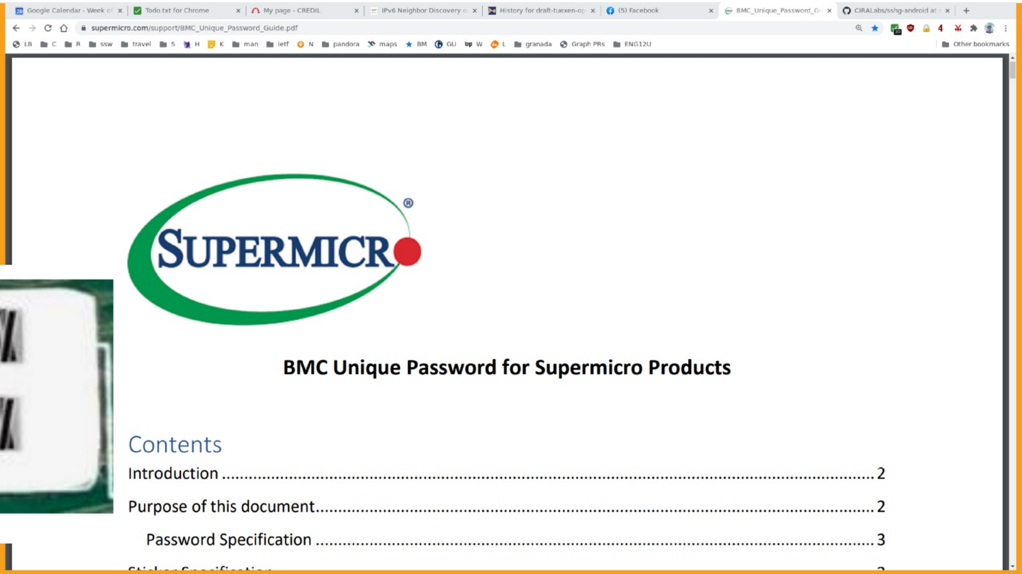
From
https://ipvm.com/reports/ip-cameras-default-passwords-directory

1. Bosch: None required, but new firmwares (6.0+) prompt users to create passwords on first login
2. Cisco: No default password, requires creation during first login
3. Dahua: Requires password creation on first login. Previously this process was recommended but could be canceled; older models default to admin/admin
4. Hanwha: admin/no default password, must be created during initial setup
5. LTS: Requires unique password creation; previously admin/12345
6. Northern: Firmware 5.3.0 and up requires unique password creation; previously Panasonic: Firmware 2.40 and up requires username/password creation; previously admin/12345
7. Pelco: New firmwares require unique password creation; previously admin/admin
8. Samsung (new): Previously admin/4321, but new firmwares require unique password creation

Some good password stories (2)





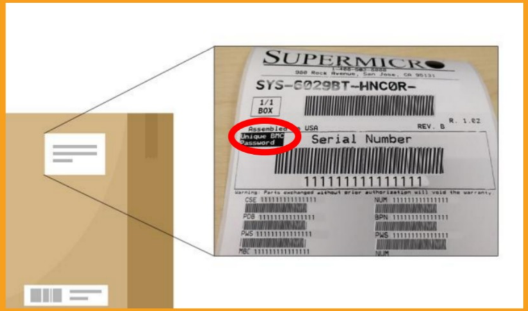


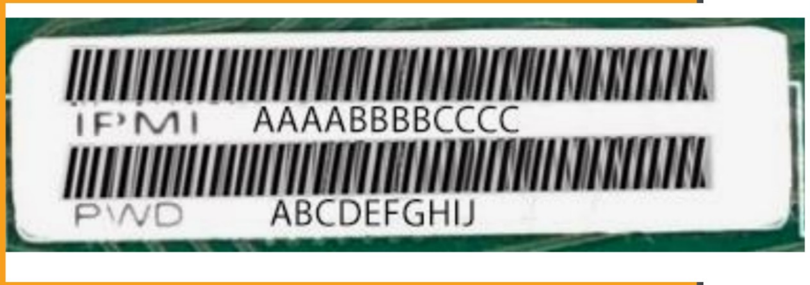
BMC Unique Password for Supermicro Products

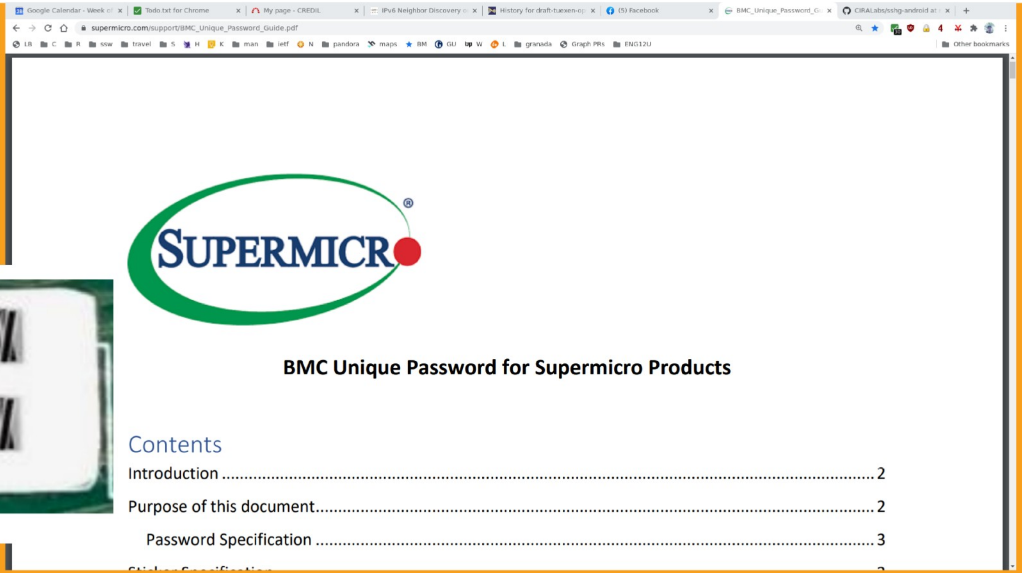
Contents

from
https://www.supermicro.com/support/BMC_Unique_Password_Guide.pdf

Some good password stories (2)







supermicro.com/support/BMC_Unique_Password_Guide.pdf

BMC Unique Password for Supermicro Products

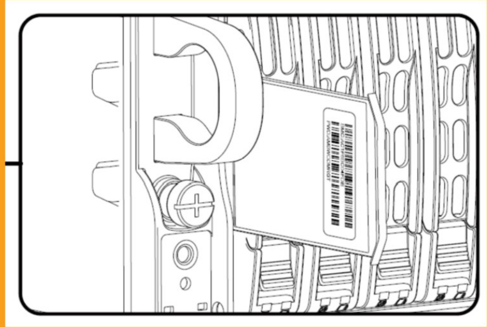
Contents

Introduction 2

Purpose of this document.. 2

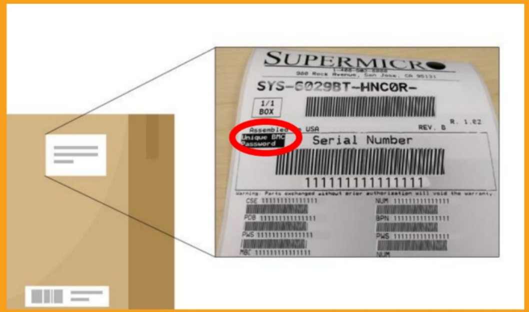
Password Specification 3

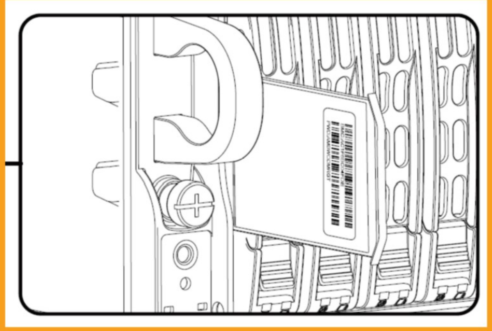
Sticker Specification 3

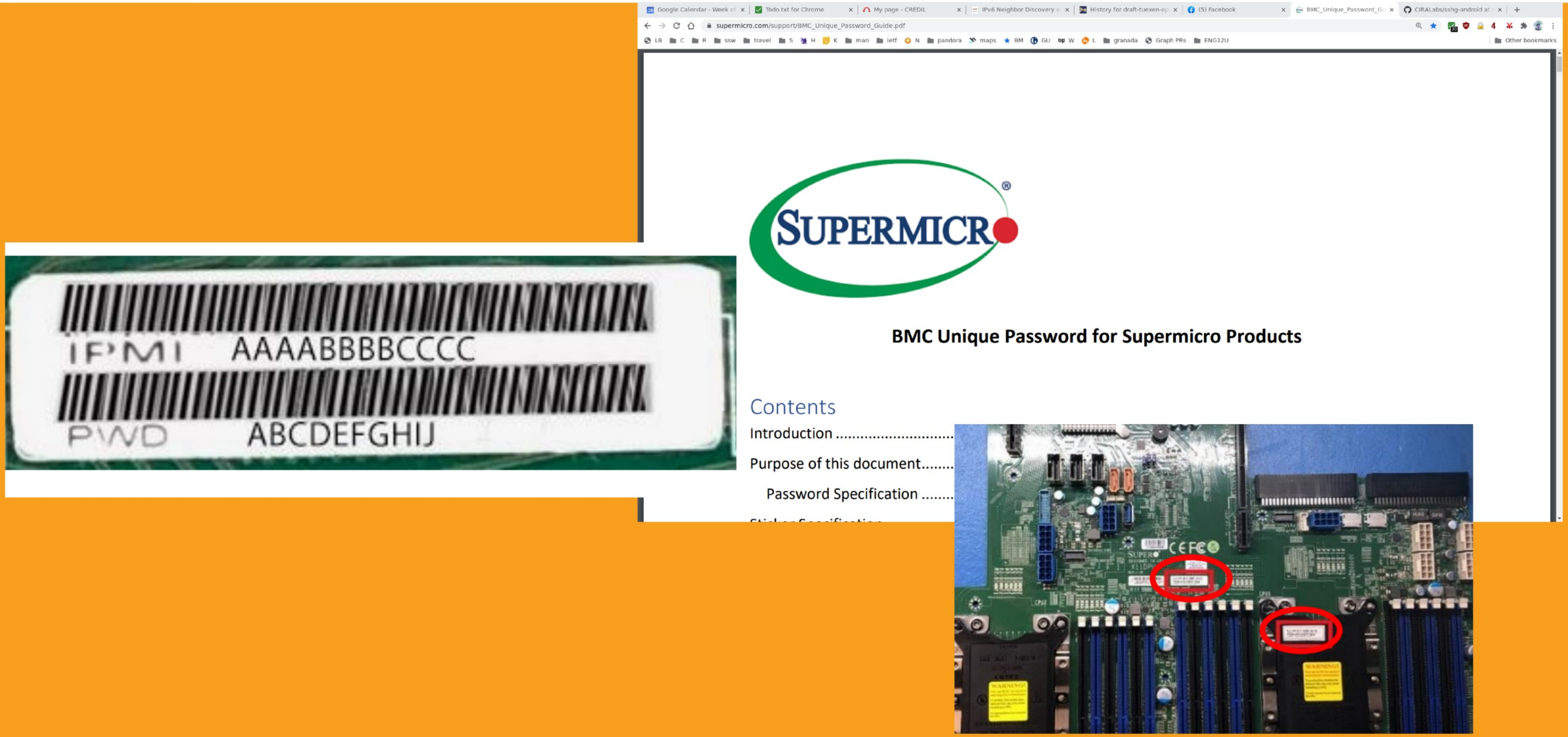


from https://www.supermicro.com/support/BMC_Unique_Password_Guide.pdf

Some good password stories (2)







BMC Unique Password for Supermicro Products

Contents

from
https://www.supermicro.com/support/BMC_Unique_Password_Guide.pdf

Questions (1)

Why are passwords used?

A: Authentication

but

usually what everyone cares about is authorization

Decision: Is this entity allowed to control the device?

Are passwords the only way?

Maybe - Depends upon context

In this decade, there is always an app.

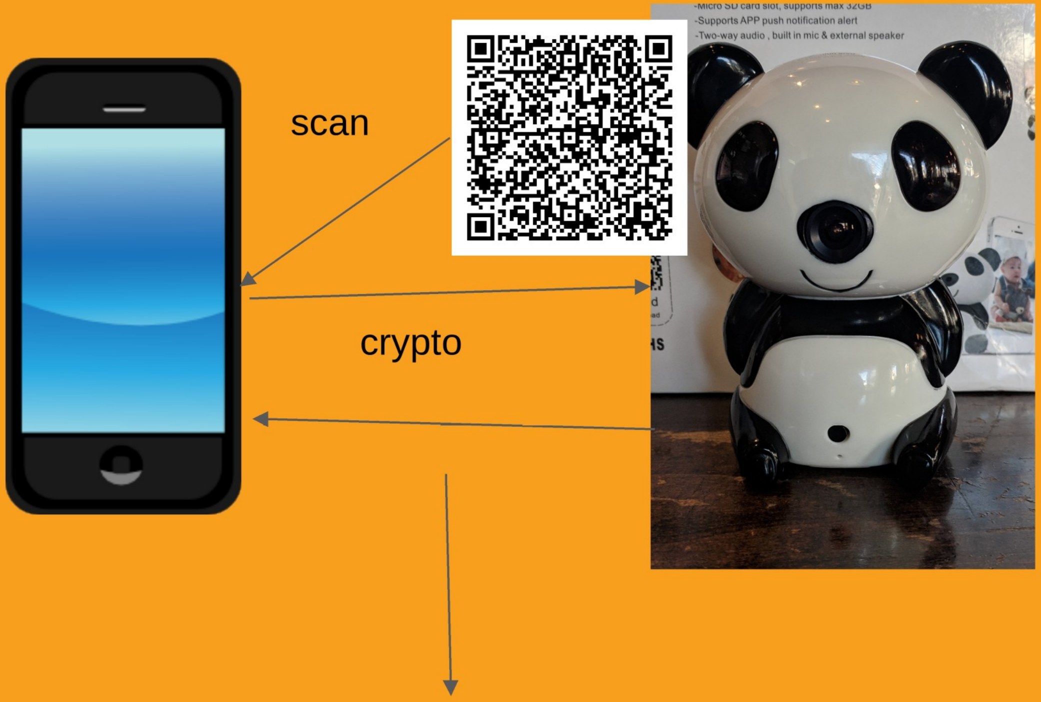
Apps do not need passwords.

Apps need cryptographic contexts:

Public keys and HTTPS/TLS

(or JSON Web Tokens: JWT)

The app: secure Onboarding provides Secure Connection


-Micro SD card slot, supports max 32GB
-Supports APP push notification alert
-Two-way audio , built in mic & external speaker

scan

crypto

Trusted relationship

Do you even need a password?

Onboarding solutions

- Wifi Alliance: EasyConnect
  - Device Provisioning Protocol (DPP)
  - https://www.wi-fi.org/discover-wi-fi/wi-fi-easy-connect
- Thread Commissioning
- https://openthread.io/guides/build/commissioning
- Amazon/Google IoT:
- https://www.trustonic.com/solutions/iot-security/automatic-cloud-enrollment/
- Intel Secure Device Onboarding (SDO)
- https://software.intel.com/content/www/us/en/develop/tools/secure-device-onboard.html

Upcoming:

- Bootstrapping Remote Secure Key Infrastructure (BRSKI) from IETF/ANIMA
- EAP-NOOB: IETF/EMU WG
- Zigbee CHIP (?) - TBD

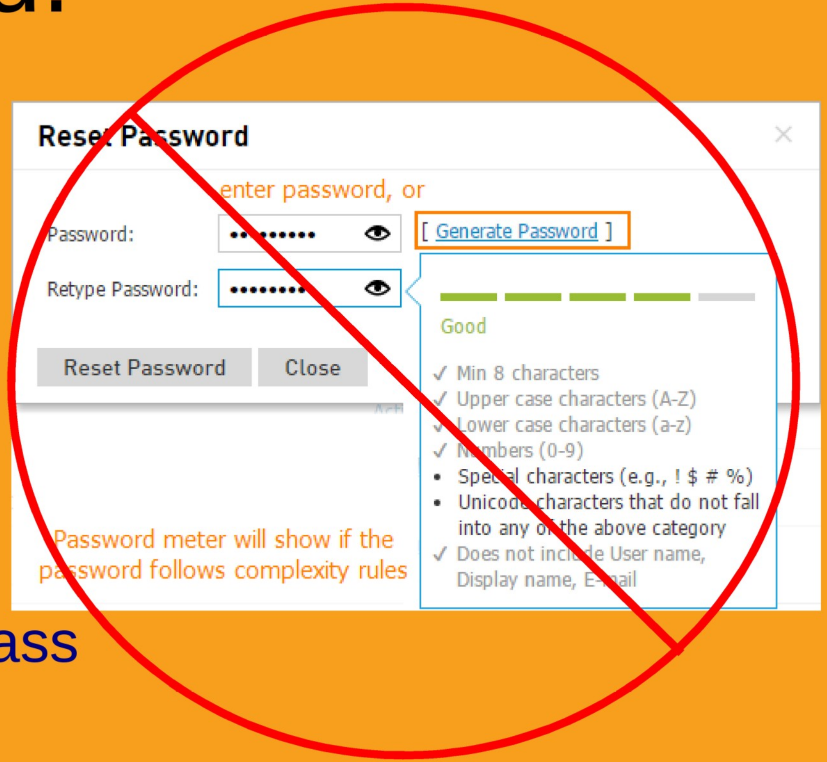
Password Best Practices

Okay. You still want a password.

NCSC guidelines and NIST guidelines

- https://www.ncsc.gov.uk/collection/passwords
- https://pages.nist.gov/800-63-3/

- No reason to make good passwords expire!
- No need for special characters, it does not help
- Encourage use of phrases
- Let users copy and paste, use password managers
  - https://www.ncsc.gov.uk/blog-post/let-them-paste-passwords



Reset Password

enter password, or

Password: ••••••••• [Generate Password]

Retype Password: •••••••••

Good

✓ Min 8 characters
✓ Upper case characters (A-Z)
✓ Lower case characters (a-z)
✓ Numbers (0-9)
• Special characters (e.g., ! $ # %)
• Unicode characters that do not fall into any of the above category
✓ Does not include User name, Display name, E-mail

Password meter will show if the password follows complexity rules

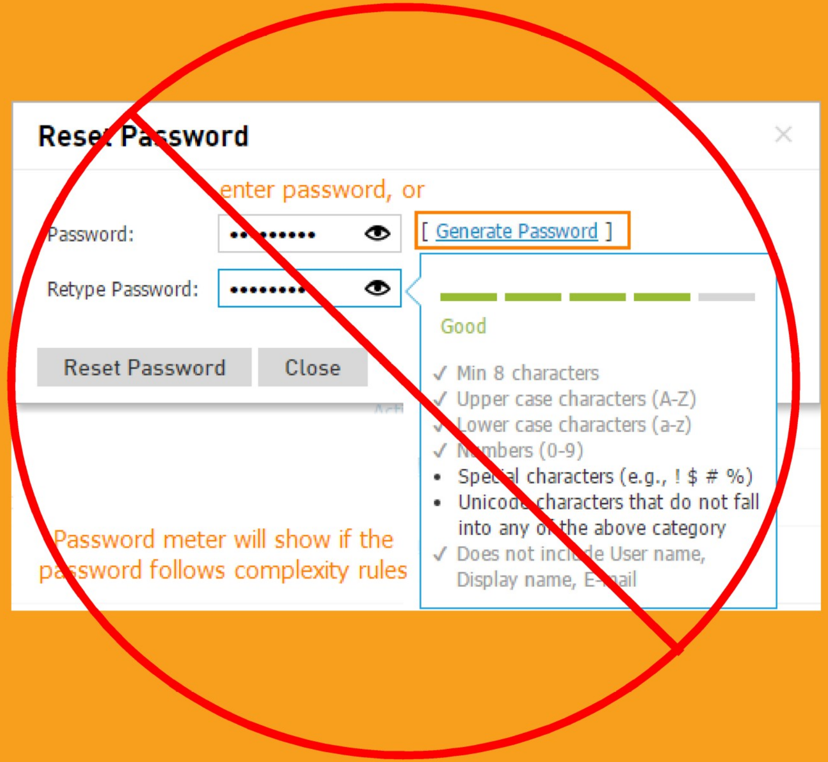
Password Best Practices

If the device is already online, then using online databases such as:

https://haveibeenpwned.com/

Is simpler than trying to build-in a list of known-bad passwords.

https://www.ncsc.gov.uk/section/advice-guidance/all-topics?topics=Passwords



Reset Password

enter password, or

Password: ••••••••• [Generate Password]

Retype Password: •••••••••

Good

- ✓ Min 8 characters
- ✓ Upper case characters (A-Z)
- ✓ Lower case characters (a-z)
- ✓ Numbers (0-9)
- • Special characters (e.g., ! $ # %)
- • Unicode characters that do not fall into any of the above category
- ✓ Does not include User name, Display name, E-mail

Reset Password | Close

Password meter will show if the password follows complexity rules

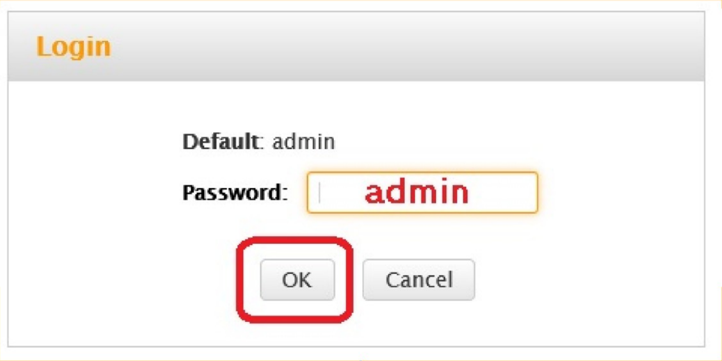
Bio-Break

Security Foundation

OXFORD INFORMATION LABS

Initial Passwords

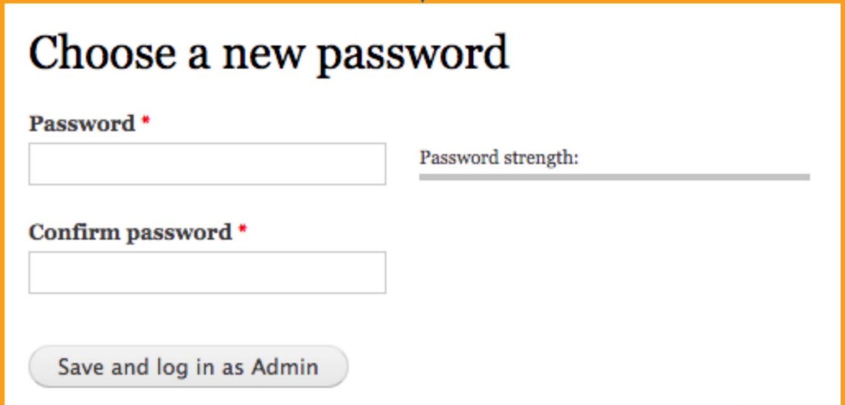
- The password that the device will accept after a factory default/reset.
-
- **LET THEM USE A PASSWORD MANAGER**
  - **https://www.ncsc.gov.uk/blog-post/let-them-paste-passwords**
- (but a lot more about initial passwords later)



Login

Default: admin

Password: admin

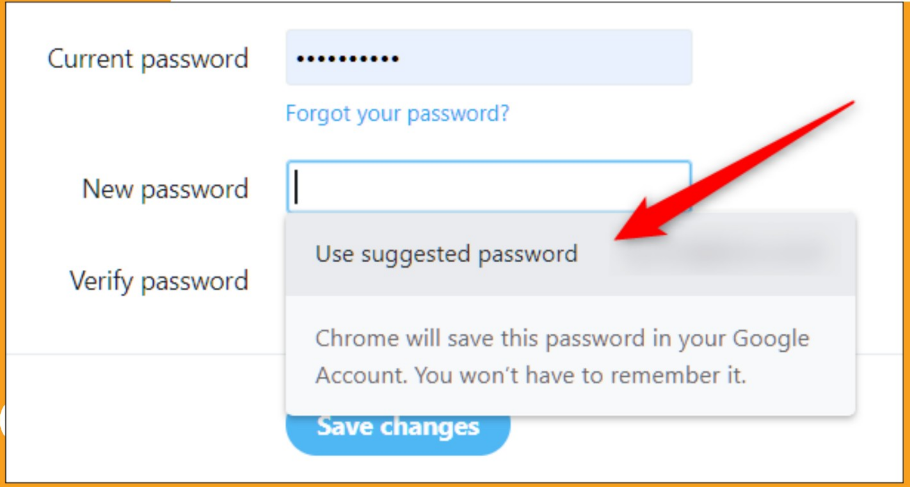


Choose a new password

Password *

Password strength:

Confirm password *



Current password ••••••••••

Forgot your password?

New password

Verify password

Use suggested password

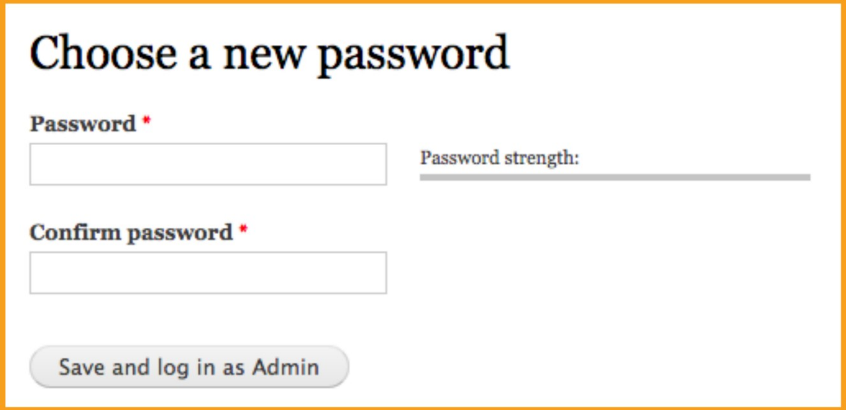
Chrome will save this password in your Google Account. You won't have to remember it.

No Initial Password

- The initial password is... NO PASSWORD
- …
- **As long as the password is changed before the device goes online.**

- The user will be forced to change it, so why even have one?
- Some services/devices send an email or SMS with a password *every time*

Must emphasize this point. Requires physical Access!



Choose a new password

Password *

Password strength:

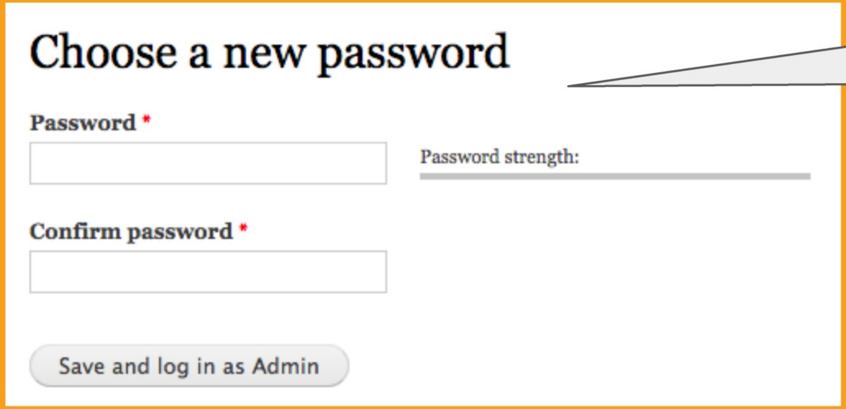
Confirm password *

No Initial Password

- The initial password is... NO PASSWORD
- …
- **As long as the password is changed before the device goes online.**

- The user will be forced to change it, so why even have one?
- Some services/devices send an email or SMS with a password *every time*

Must emphasize this point. Requires physical Access!



Choose a new password

Password *

Password strength:

Confirm password *

From a newly installed Drupal Content Management System

A problem with no, or weak initial password

- The user will be *forced to change it*, so why even have one?

A problem with no, or weak initial password

- The user will be *forced to change it*, so why even have one?

What if they never use the “smart” aspect of the device?
They device remains uninitialized?

A problem with no, or weak initial password

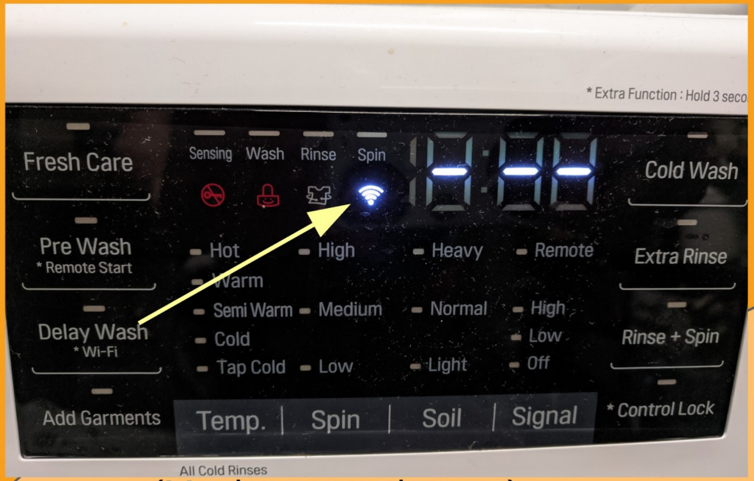
- The user will be *forced to change it*, so why even have one?

What if they never use the “smart” aspect of the device?
They device remains uninitialized?

(Maybe not webcams)
Refridgerators, clothes washers, dryers, stoves, microwaves, TVs, garage openers, even “smart” locks

A problem with no, or weak initial password

- The user will be *forced to change it*, so why even have one?



What if they never use the “smart” aspect of the device?
They device remains uninitialized?

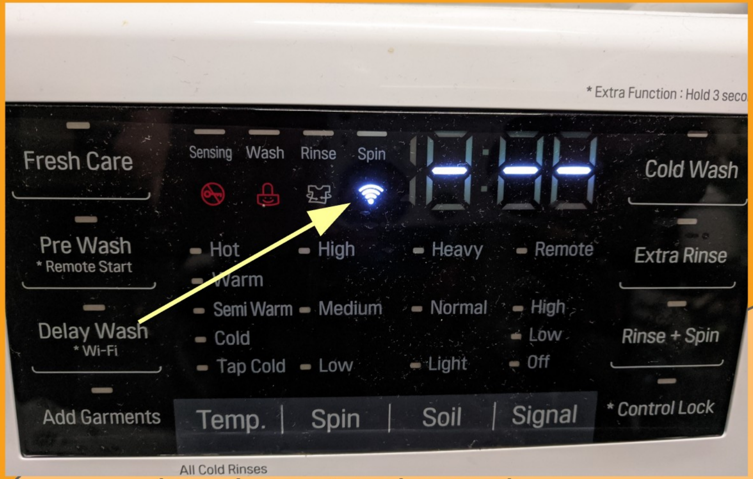
(Maybe not webcams)
Refridgerators, clothes washers, dryers, stoves, microwaves, TVs, garage openers, even “smart” locks

IoT Security Foundation

OXFORD INFORMATION LABS

A problem with no, or weak initial password

- The user will be *forced to change it*, so why even have one?



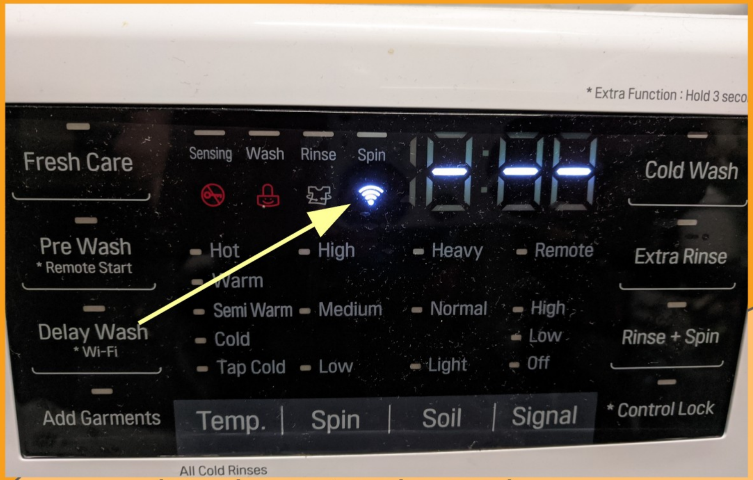
What if they never use the “smart”
aspect of the device?
They device remains uninitialized?

(Maybe not webcams)
Refridgerators, clothes washers, dryers,
stoves, microwaves, TVs, garage openers,
even “smart” locks

Device must remain
completely *inoperable* and entirely
safe, lest war-driving
attacker take over the device!

A problem with no, or weak initial password

- The user will be *forced to change it*, so why even have one?



What if they never use the "smart" aspect of the device?
They device remains uninitialized?

(Maybe not webcams)
Refridgerators, clothes washers, dryers, stoves, microwaves, TVs, garage openers, even "smart" locks

Device must remain completely *inoperable* and entirely *safe*, lest war-driving attacker take over the device!

If device is online by default, strong initial passwords are still needed!!

Storing and using passwords

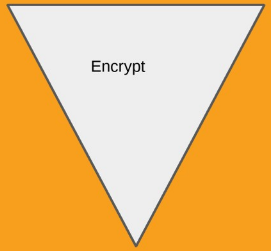
All password need to be stored in some way so that a correct password is recognized.

- Usually, it is poor practice to store the password un-encrypted, as it could possibly be retrieved by physical attacks, or through software bugs.
- The traditional Unix/POSIX method is to use the password to encrypt or hash some value, and then compare the result.

Storing and using passwords

All password need to be stored in some way so that a correct password is recognized.

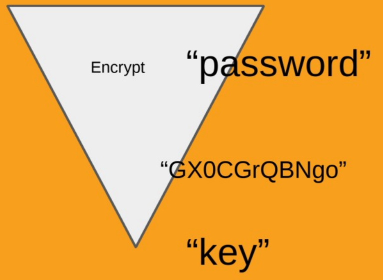
- Usually, it is poor practice to store the password un-encrypted, as it could possibly be retrieved by physical attacks, or through software bugs.
- The traditional Unix/POSIX method is to use the password to encrypt or hash some value, and then compare the result.



Storing and using passwords

All password need to be stored in some way so that a correct password is recognized.

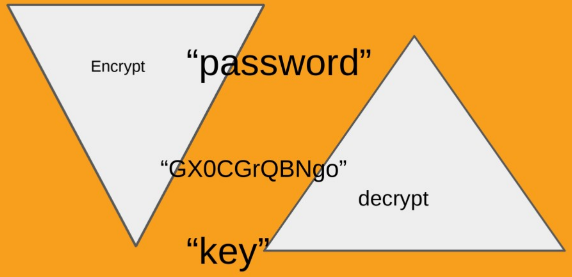
- Usually, it is poor practice to store the password un-encrypted, as it could possibly be retrieved by physical attacks, or through software bugs.
- The traditional Unix/POSIX method is to use the password to encrypt or hash some value, and then compare the result.



Storing and using passwords

All password need to be stored in some way so that a correct password is recognized.

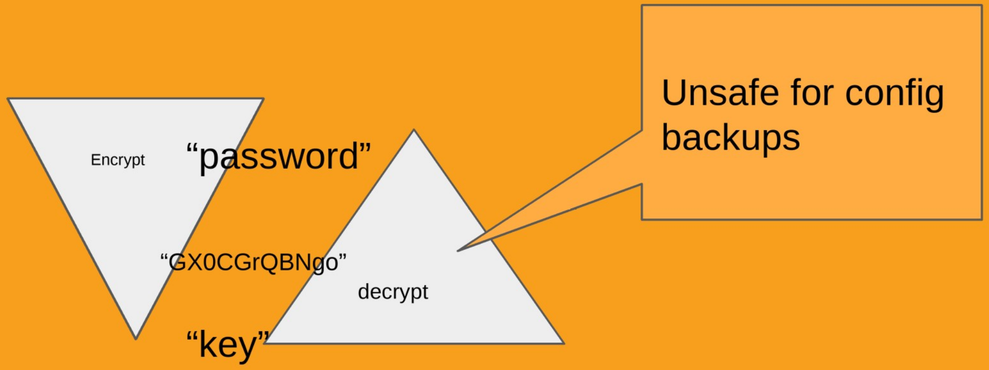
- Usually, it is poor practice to store the password un-encrypted, as it could possibly be retrieved by physical attacks, or through software bugs.
- The traditional Unix/POSIX method is to use the password to encrypt or hash some value, and then compare the result.



Storing and using passwords

All password need to be stored in some way so that a correct password is recognized.

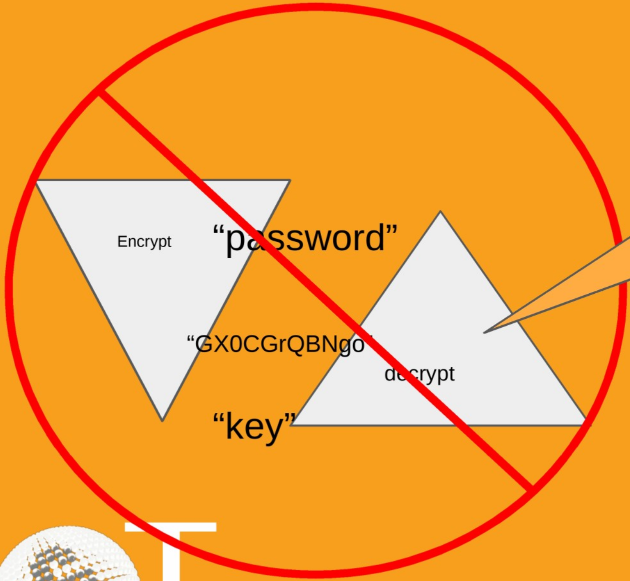
- Usually, it is poor practice to store the password un-encrypted, as it could possibly be retrieved by physical attacks, or through software bugs.
- The traditional Unix/POSIX method is to use the password to encrypt or hash some value, and then compare the result.



Storing and using passwords

All password need to be stored in some way so that a correct password is recognized.

- Usually, it is poor practice to store the password un-encrypted, as it could possibly be retrieved by physical attacks, or through software bugs.
- The traditional Unix/POSIX method is to use the password to encrypt or hash some value, and then compare the result.

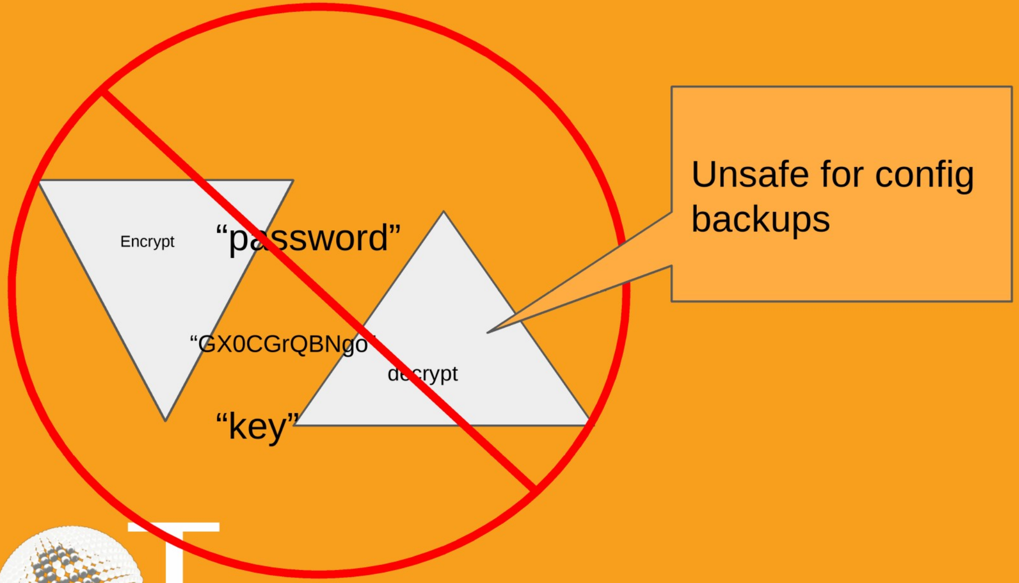


Unsafe for config backups

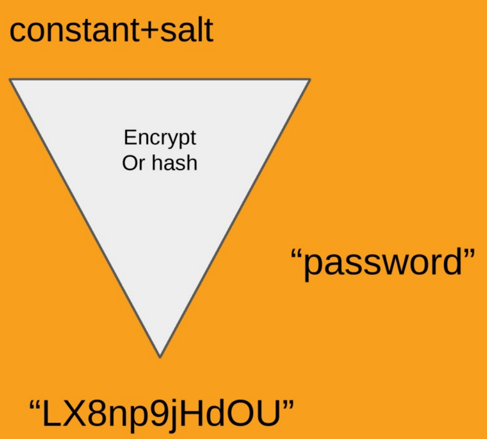
Storing and using passwords

All password need to be stored in some way so that a correct password is recognized.

- Usually, it is poor practice to store the password un-encrypted, as it could possibly be retrieved by physical attacks, or through software bugs.
- The traditional Unix/POSIX method is to use the password to encrypt or hash some value, and then compare the result.



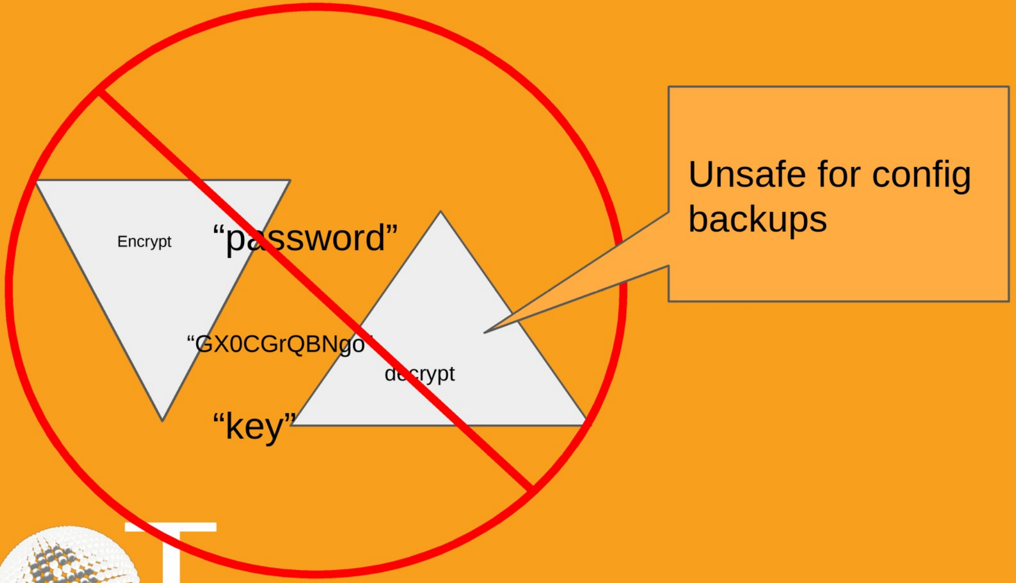
Unsafe for config backups



Storing and using passwords

All password need to be stored in some way so that a correct password is recognized.

- Usually, it is poor practice to store the password un-encrypted, as it could possibly be retrieved by physical attacks, or through software bugs.
- The traditional Unix/POSIX method is to use the password to encrypt or hash some value, and then compare the result.



Encrypt

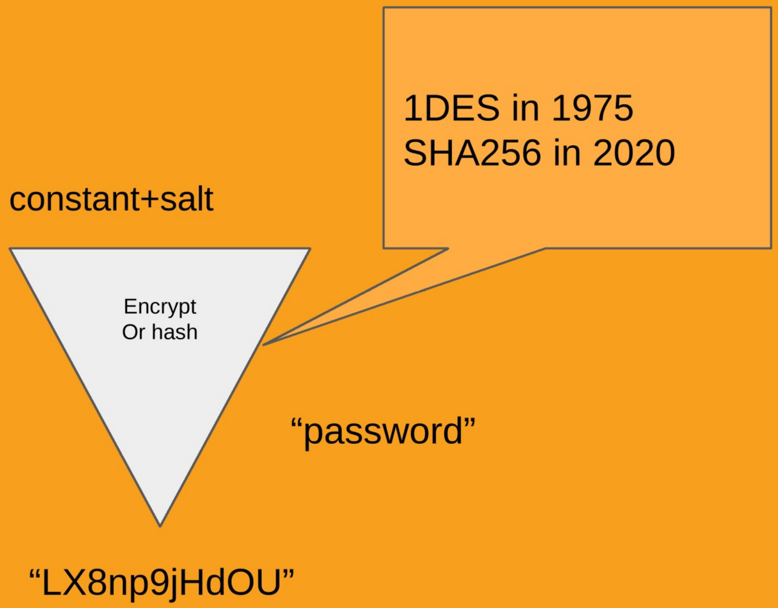
“password”

“GX0CGrQBNgo”

decrypt

“key”

Unsafe for config backups



1DES in 1975
SHA256 in 2020

constant+salt

Encrypt
Or hash

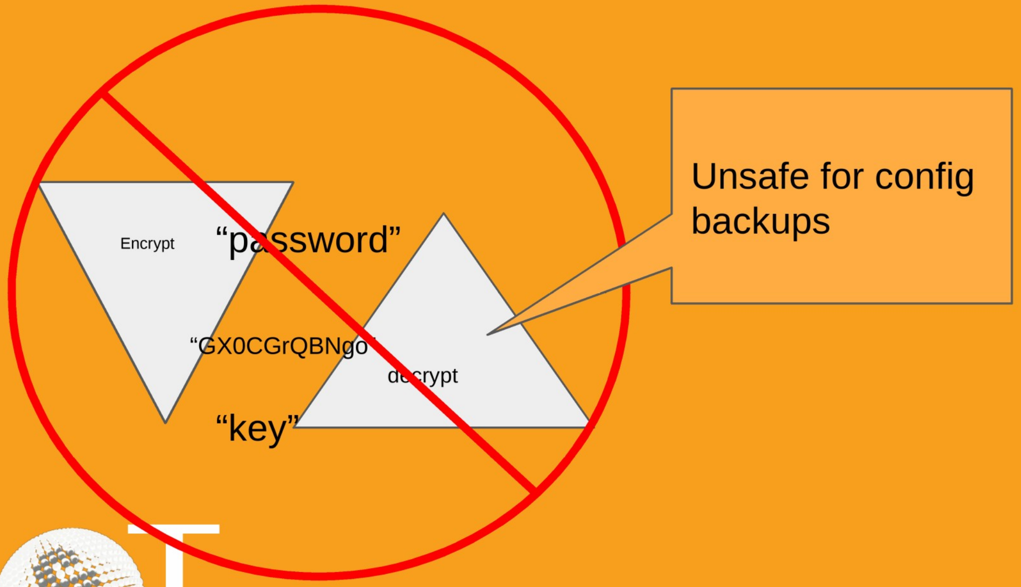
“password”

“LX8np9jHdOU”

Storing and using passwords

All password need to be stored in some way so that a correct password is recognized.

- Usually, it is poor practice to store the password un-encrypted, as it could possibly be retrieved by physical attacks, or through software bugs.
- The traditional Unix/POSIX method is to use the password to encrypt or hash some value, and then compare the result.



Encrypt

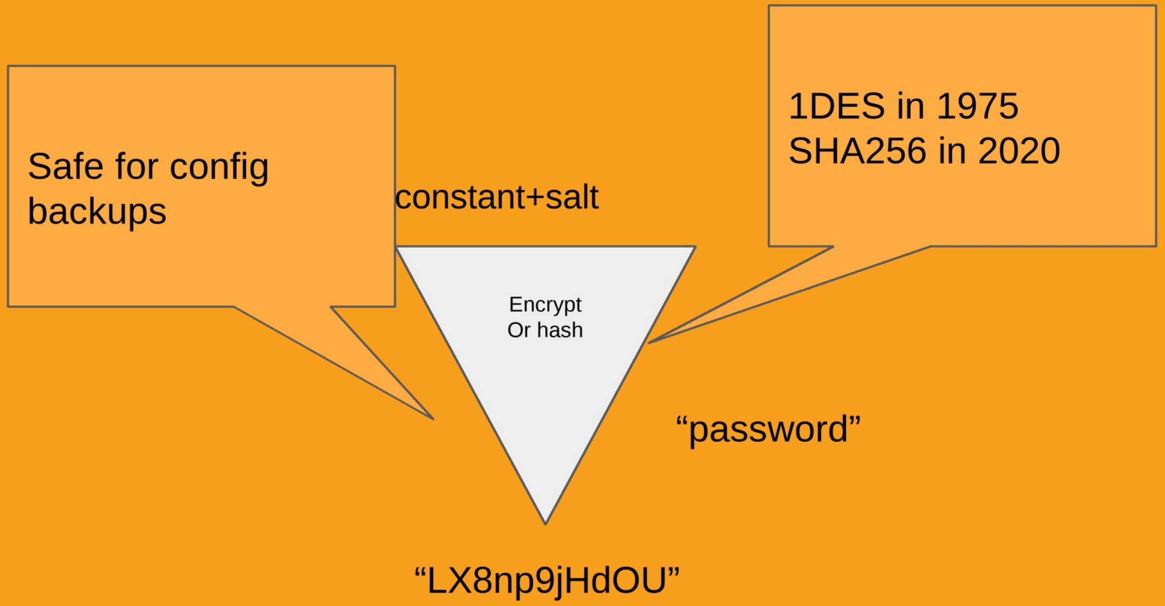
“password”

“GX0CGrQBNgo”

decrypt

“key”

Unsafe for config backups



Safe for config backups

constant+salt

1DES in 1975
SHA256 in 2020

Encrypt
Or hash

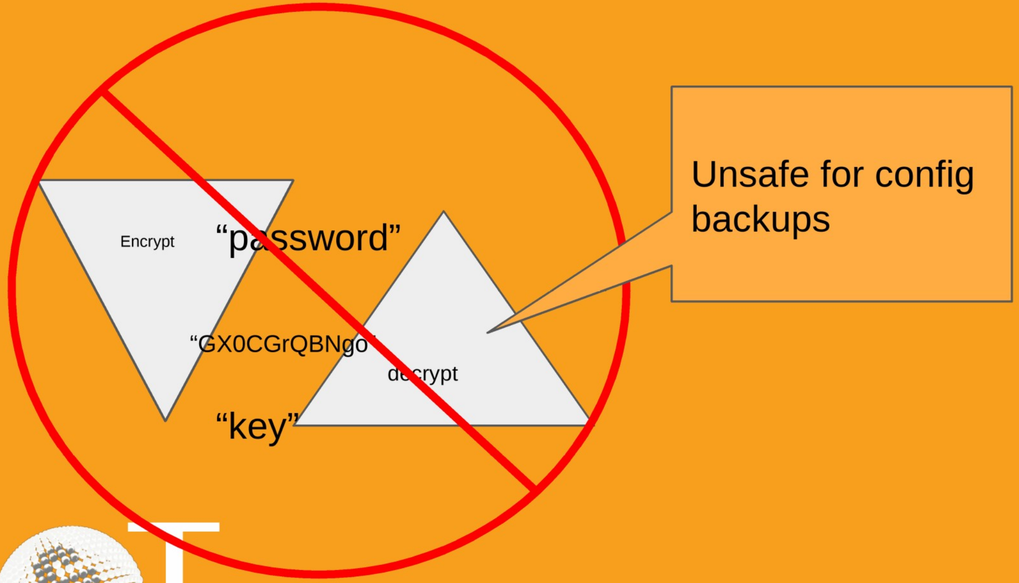
“password”

“LX8np9jHdOU”

Storing and using passwords

All password need to be stored in some way so that a correct password is recognized.

- Usually, it is poor practice to store the password un-encrypted, as it could possibly be retrieved by physical attacks, or through software bugs.
- The traditional Unix/POSIX method is to use the password to encrypt or hash some value, and then compare the result.



Encrypt

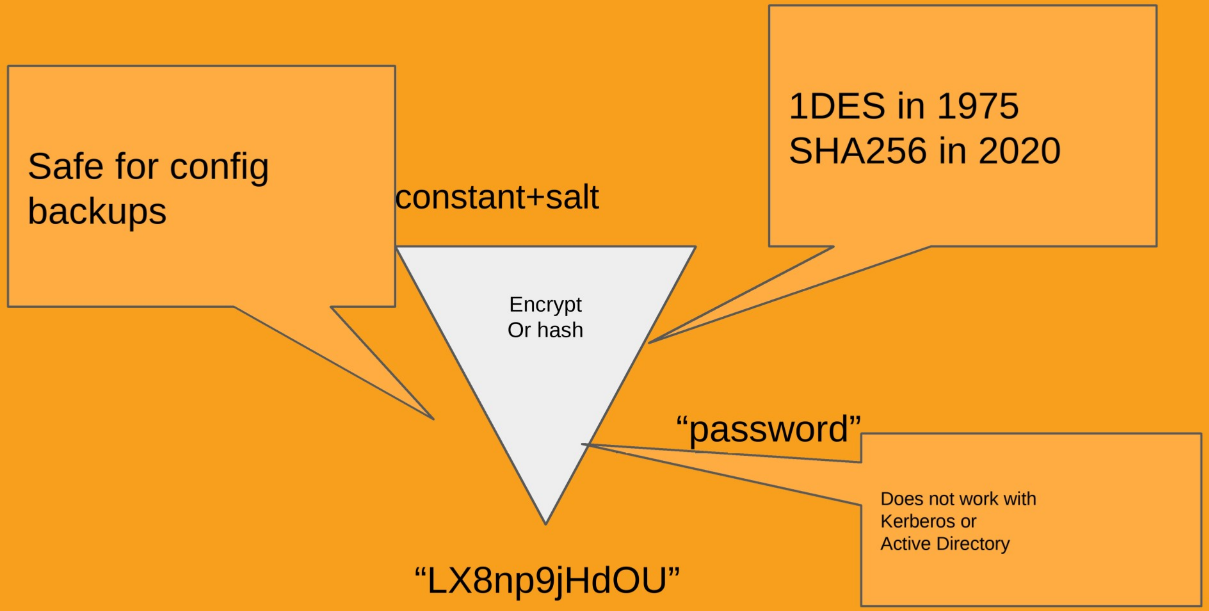
“password”

“GX0CGrQBNgo”

decrypt

“key”

Unsafe for config backups



Safe for config backups

constant+salt

1DES in 1975
SHA256 in 2020

Encrypt
Or hash

“password”

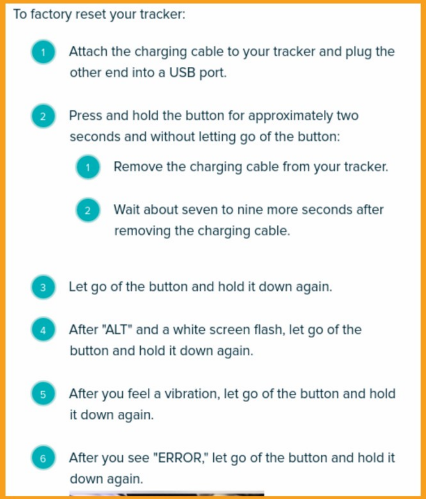
Does not work with
Kerberos or
Active Directory

“LX8np9jHdOU”

Recovering lost passwords

What if they user forgets the password?

- Is there a cloud service with their email? Then use that.
    - Follow password reset advice: https://postmarkapp.com/guides/password-reset-email-best-practices
- Is there integration with social media? Then maybe use that.
- Does the device have a screen/buttons? Then use that.
    - But, is the device physically secure? Like a washing machine?
    - Or is it portable, like a fitness monitor, or shared item like gym equipment, or used in schools?
        - Tie password resets to device communication, so fitness device loses pairing if reset.
        - That way, legitimate owner will have to factory reset it again: annoying, but secure



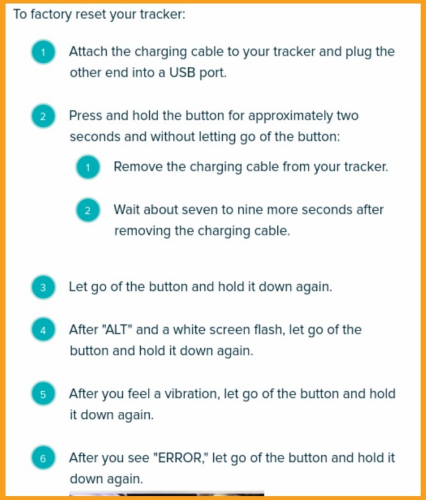
To factory reset your tracker:

1. Attach the charging cable to your tracker and plug the other end into a USB port.
2. Press and hold the button for approximately two seconds and without letting go of the button:
    1. Remove the charging cable from your tracker.
    2. Wait about seven to nine more seconds after removing the charging cable.
3. Let go of the button and hold it down again.
4. After "ALT" and a white screen flash, let go of the button and hold it down again.
5. After you feel a vibration, let go of the button and hold it down again.
6. After you see "ERROR," let go of the button and hold it down again.

Recovering lost passwords

Factory Reset for Passwords

- Ultimately the factory reset process will get used.
  - Factory reset can not be made too hard.
  - Can not be too easy either.
- Consider seniors, and differently abled: can they press all four required buttons?
  - (my mom can’t due to vision, Parkensons, …)



To factory reset your tracker:

1. Attach the charging cable to your tracker and plug the other end into a USB port.
2. Press and hold the button for approximately two seconds and without letting go of the button:
   1. Remove the charging cable from your tracker.
   2. Wait about seven to nine more seconds after removing the charging cable.
3. Let go of the button and hold it down again.
4. After "ALT" and a white screen flash, let go of the button and hold it down again.
5. After you feel a vibration, let go of the button and hold it down again.
6. After you see "ERROR," let go of the button and hold it down again.

Bruce Force Attacks on passwords

A brute force attack is one where the attacker simply tries a lot of combinations: Trillions and Trillions.

There are two kinds of Brute Force attacks:

1. Online attacks against the device itself
2. Offline attacks against a copy of the configuration or firmware

Bruce Force Attacks on passwords

A brute force attack is one where the attacker simply tries a lot of combinations: Trillions and Trillions.

There are two kinds of Brute Force attacks:

1. Online attacks against the device itself
2. Offline attacks against a copy of the configuration or firmware

Online attacks require the device to do some work.

The device is in charge of how fast it will do the work.

The best defense is to be **slow**.

But, be slow in a *constant* way, so that all failures take the same amount of time to avoid differential/timing attacks.

Bruce Force Attacks on passwords

A brute force attack is one where the attacker simply tries a lot of combinations: Trillions and Trillions.

There are two kinds of Brute Force attacks:

1. Online attacks against the device itself
2. Offline attacks against a copy of the configuration or firmware

Online attacks require the device to do some work.

The device is in charge of how fast it will do the work.

The best defense is to be **slow**.

But, be slow in a *constant* way, so that all failures take the same amount of time to avoid differential/timing attacks.

Offline attacks are done by the attacker using their own (faster) equipment.

The attacker can apply as much effort as they like, including buying cloud resources, or even stealing them.

The best defense is to never have a secret!

Assume the attacker has a copy of your source code. Design appropriately. This is why encrypt/decrypt for passwords is a bad idea.

Bruce Force Attacks on passwords

A brute force attack is one where the attacker simply tries a lot of combinations: Trillions and Trillions.

There are two kinds of Brute Force attacks:

1. Online attacks against the device itself
2. Offline attacks against a copy of the configuration or firmware

https://en.wikipedia.org/wiki/EFF_DES_cracker

Online attacks require the device to do some work.

The device is in charge of how fast it will do the work.

The best defense is to be **slow**.

But, be slow in a *constant* way, so that all failures take the same amount of time to avoid differential/timing attacks.

Offline attacks are done by the attacker using their own (faster) equipment.

The attacker can apply as much effort as they like, including buying cloud resources, or even stealing them.

The best defense is to never have a secret!

Assume the attacker has a copy of your source code. Design appropriately. This is why encrypt/decrypt for passwords is a bad idea.

Questions (2)

Provisioning (good) passwords

- How bad passwords are provisioned
- Password generated by factory, installed by JTAG
- Password generated locally, retrieved by JTAG
- Password generated, uploaded by secure HTTPS
- Password co-generated from silicon provisioned secret
  - Physically Unclonnable Function (PUF), Silicon Root of Trust, Intel SDO, ARM Pelion
  - “device-factory-co-generated-password”
- Password co-generated from OEM pseudo-secret + semi-public information
  - “device-oemhardcoded-co-generated-password”

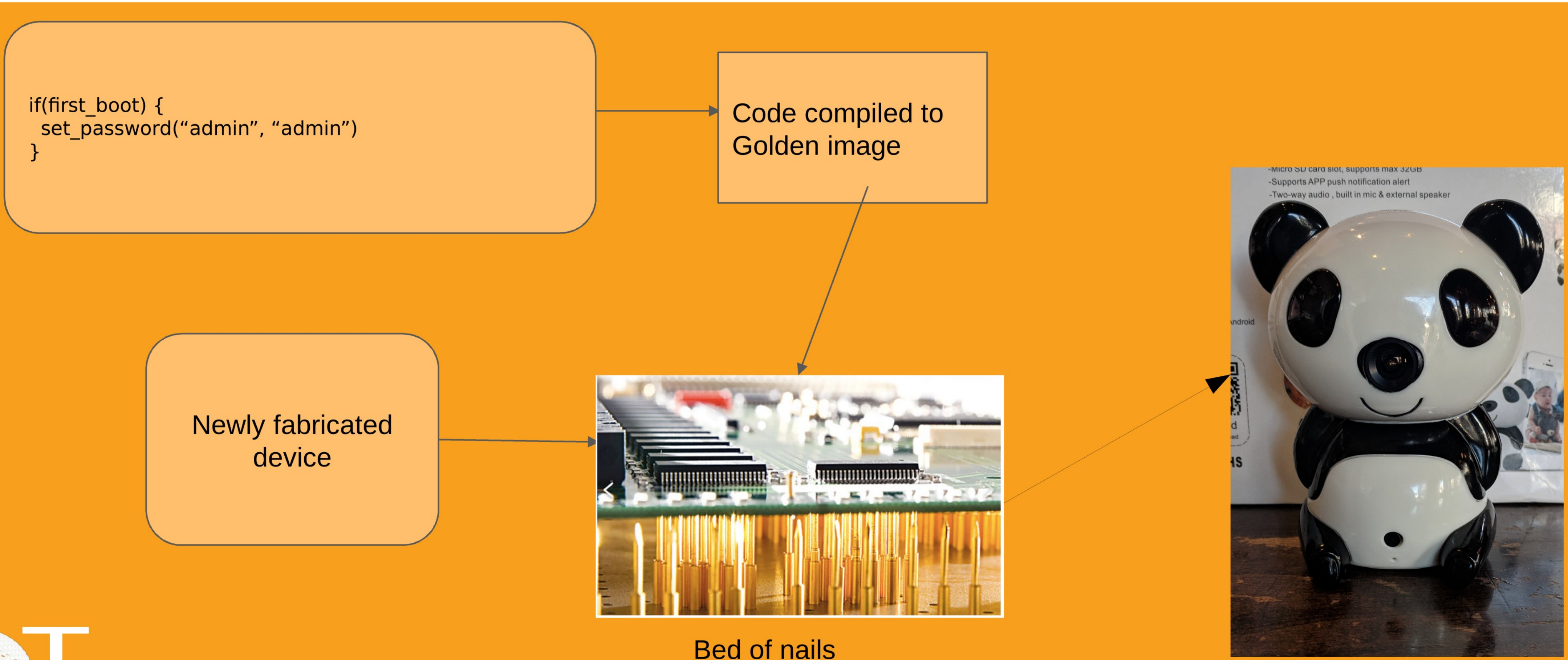
“oem-hardcoded-password”

“infrastructure-generated-password-mechanically-installed”

“device-generated-password-mechanically-retrieved”

“device-generated-password-network-retrieved”

How bad passwords are provisioned



```
if(first_boot) {
  set_password(“admin”, “admin”)
}
```

Code compiled to Golden image

Newly fabricated device

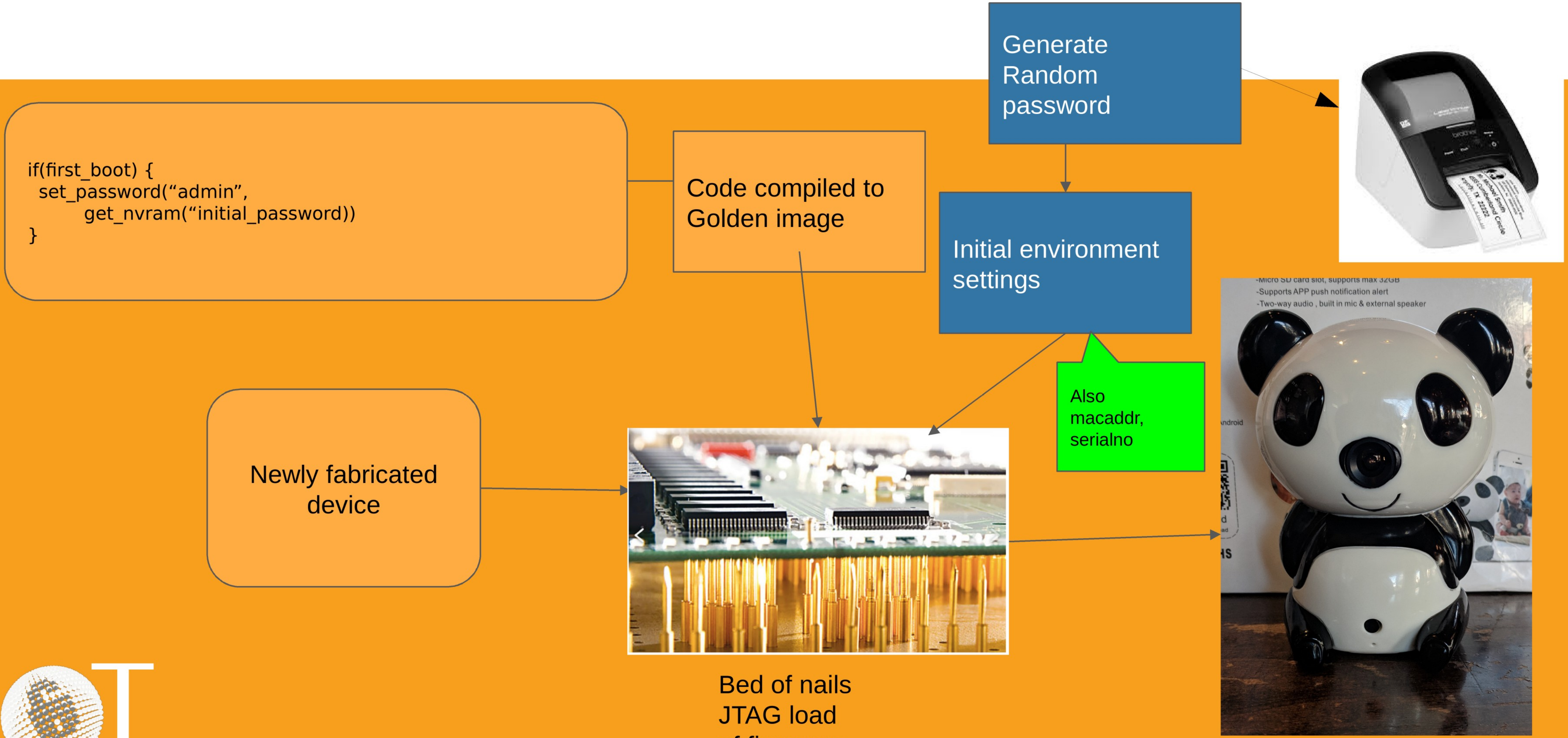
Bed of nails
JTAG load
of firmware

-Micro SD card slot, supports max 32GB
-Supports APP push notification alert
-Two-way audio , built in mic & external speaker

“factory-generated-password-mechanically-installed”

Simplest way to provision good passwords

Password installed by JTAG



```
if(first_boot) {
  set_password(“admin”,
          get_nvram(“initial_password))
}
```

Code compiled to Golden image

Generate Random password

Initial environment settings

Also macaddr, serialno

Newly fabricated device

Bed of nails
JTAG load
of firmware

-Micro SD card slot, supports max 32GB
-Supports APP push notification alert
-Two-way audio , built in mic & external speaker

"device-generated-password-mechanically-retrieved"

Another simple way to provision good passwords

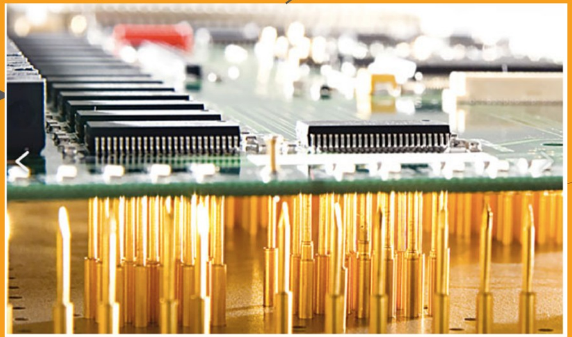
Password generated locally, retrieved by JTAG

```
if(first_boot) {
pw=generate_rnd_passwd();
set_password("admin",pw)
}
```

Code compiled to Golden image

Also macaddr, serialno

Newly fabricated device



Bed of nails
JTAG load
of firmware

Read self-test Results + PW

-Micro SD card slot, supports max 32GB
-Supports APP push notification alert
-Two-way audio , built in mic & external speaker

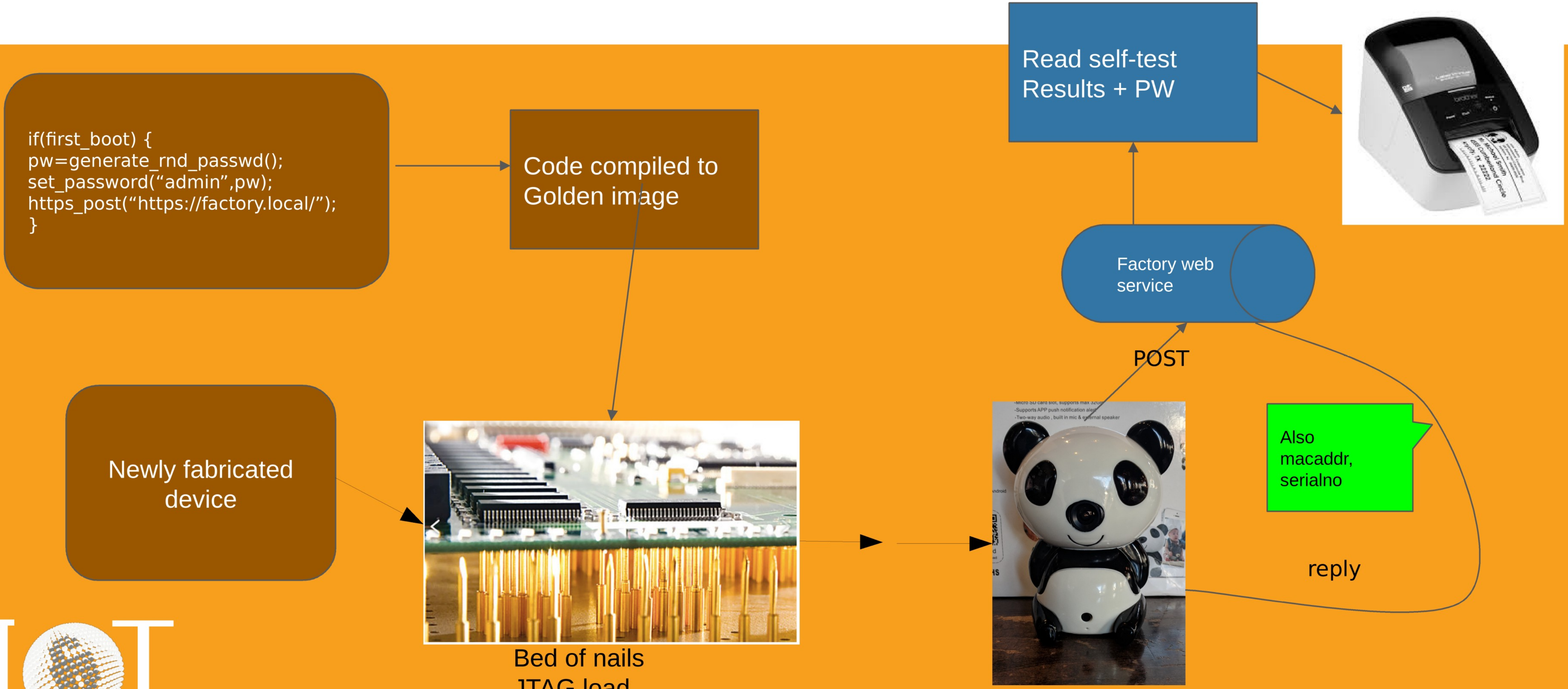
IoT Security Foundation

OXFORD INFORMATION LABS

“device-generated-password-network-retrieved”

A less simple way to provision good passwords

Password generated, uploaded by secure HTTPS



```
if(first_boot) {
pw=generate_rnd_passwd();
set_password(“admin”,pw);
https_post(“https://factory.local/”);
}
```

Code compiled to Golden image

Newly fabricated device

Bed of nails
JTAG load
of firmware

POST

Factory web service

Read self-test Results + PW

reply

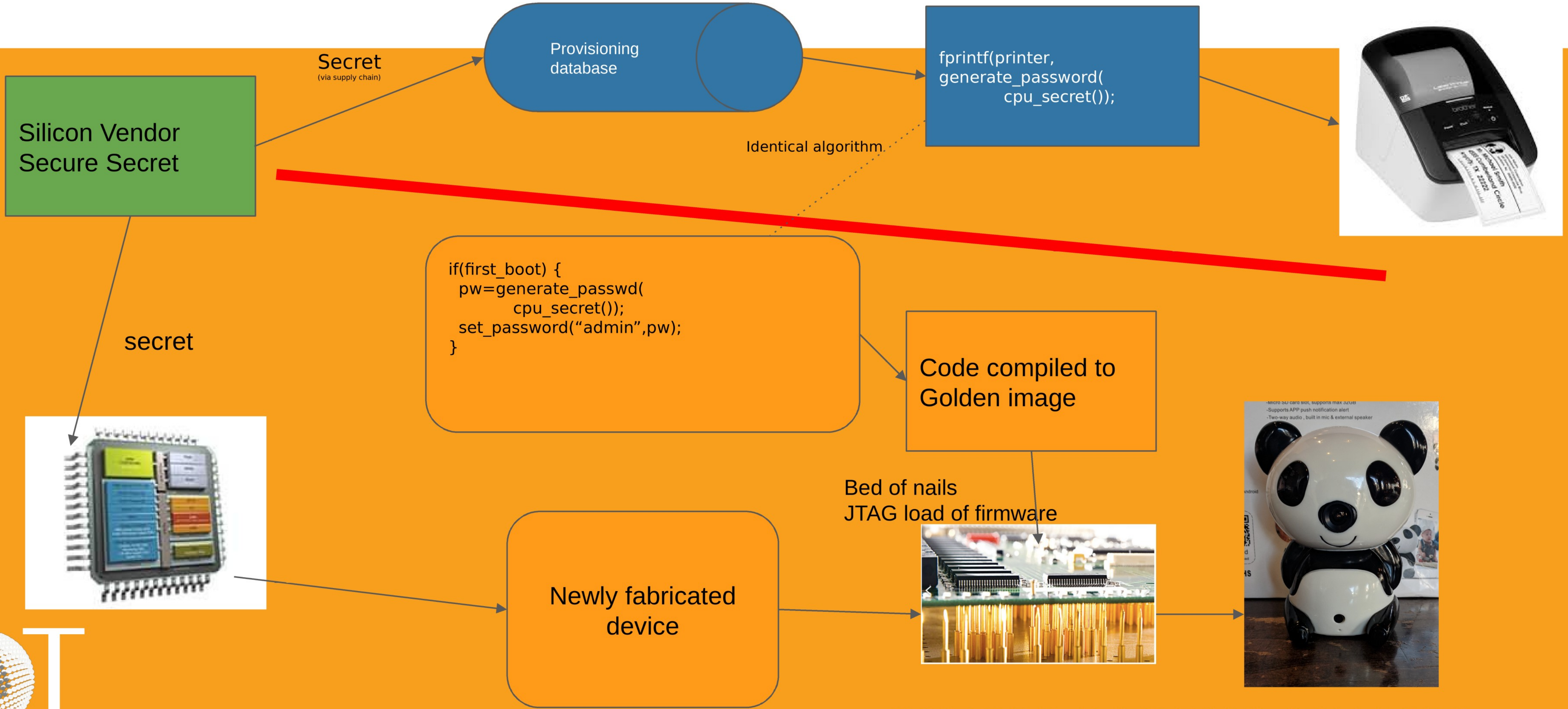
Also macaddr, serialno

IoT Security Foundation

OXFORD INFORMATION LABS

“device-factory-co-generated-password”

Leveraging CPU provisioned secrets: Password co-generated from silicon provisioned secret



Silicon Vendor Secure Secret

Secret
(via supply chain)

Provisioning database

```
fprintf(printer,
generate_password(
          cpu_secret());
```

Identical algorithm

```
if(first_boot) {
  pw=generate_passwd(
          cpu_secret());
  set_password(“admin”,pw);
}
```

Code compiled to Golden image

secret

Newly fabricated device

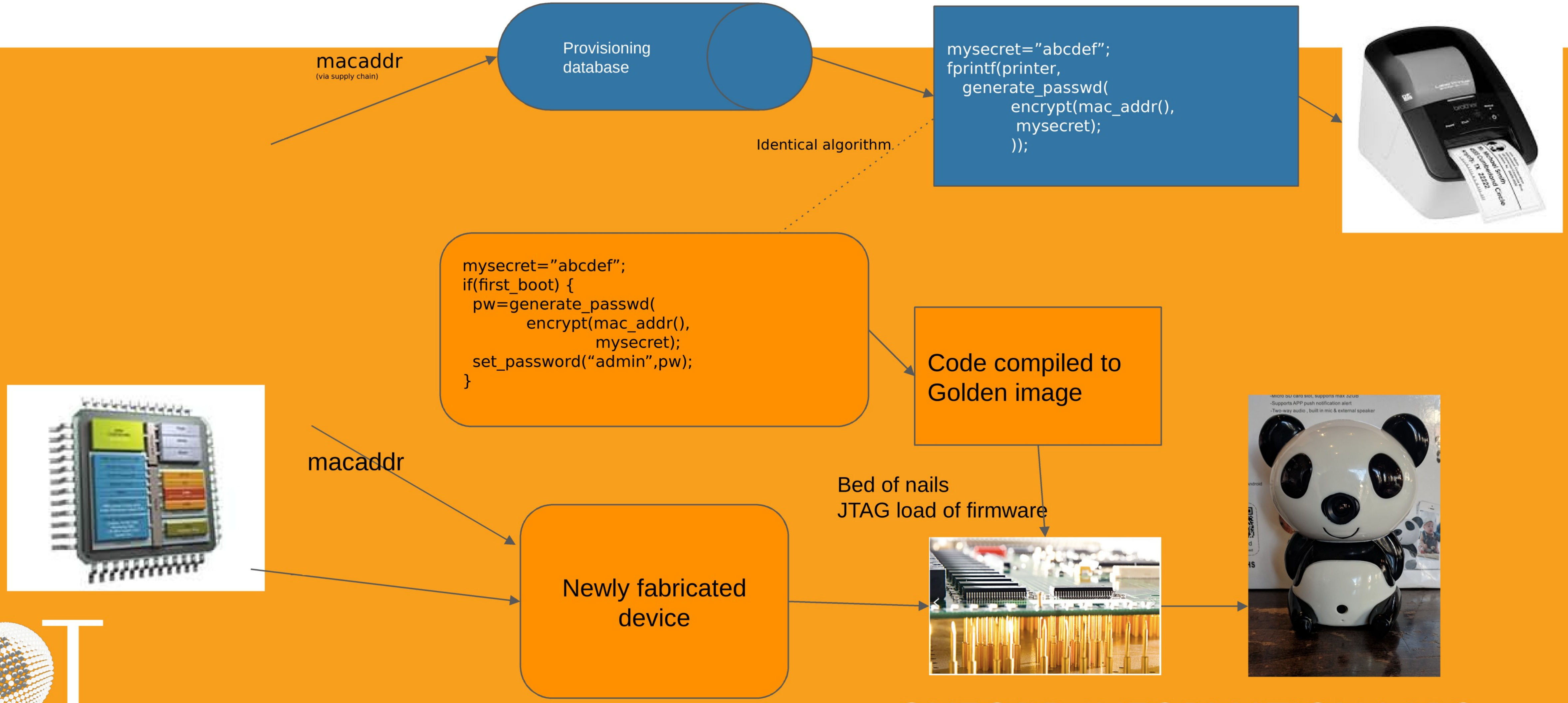
Bed of nails
JTAG load of firmware

IoT Security Foundation

OXFORD INFORMATION LABS

“device-oemhardcoded-co-generated-password”

Leveraging CPU provisioned secrets
Password co-generated from OEM pseudo-secret + semi-public information



```
mysecret=”abcdef”;
fprintf(printer,
   generate_passwd(
            encrypt(mac_addr(),
             mysecret);
            ));
```

```
mysecret=”abcdef”;
if(first_boot) {
  pw=generate_passwd(
             encrypt(mac_addr(),
                          mysecret);
  set_password(“admin”,pw);
}
```

Comparison of Methods of provisioning passwords

	oem-hardcoded-password	infrastructure-generated-password-mechanically-installed	device-generated-password-mechanically-retrieved	device-generated-password-network-retrieved	device-factory-co- generated-password	device- oem hardcoded- co-generated-password
guessable by attackers	YES	NO	NO	NO	No	YES
stored in database	N/A	YES	NO	YES	YES	NO
sensitive to RNG in device	NO	NO	YES	NO	NO	NO
sensitive to RNG in factory	NO	YES	NO	NO	YES	NO
trust of silicon vendor	NO	NO	No	NO	YES	NO

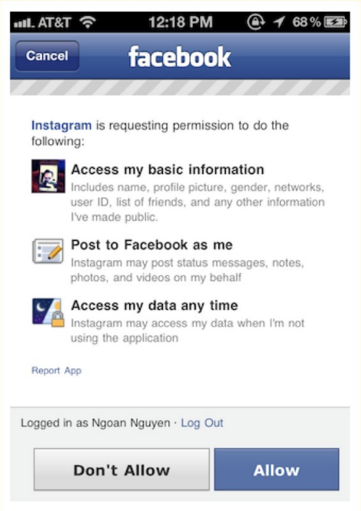
Questions (3)

Alternatives to Passwords

The onboarding processes mentioned earlier are usually used to establish WIFI credentials into a device.

But, as they result in a secure connection to the device from an app in a SmartPhone, they can also be leveraged to create a secure session. The specific way to do this is often specific to the way the device is used and controlled.   The next few slides are high-level views of:

1. Self-signed certificates, and Raw Public Keys.
2. OAUTH2, OpenID-Connect: RFC6749
3. JSON Web Tokens. (RFC7515: JWT), and also CBOR Web Token (RFC8392: CWT)
4. Authorization for Constrained Environments (ACE: RFC7744, https://datatracker.ietf.org/wg/ace/documents/)
5. Magic URLs sent via email or SMS



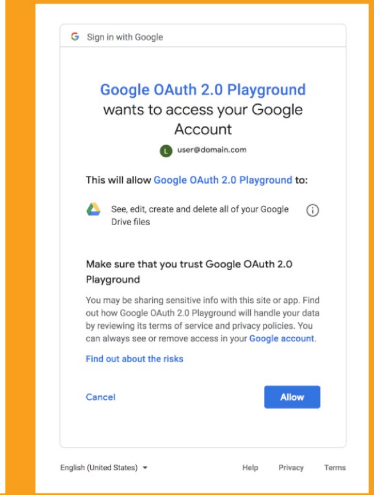
Instagram is requesting permission to do the following:

Access my basic information
Includes name, profile picture, gender, networks, user ID, list of friends, and any other information I've made public.

Post to Facebook as me
Instagram may post status messages, notes, photos, and videos on my behalf

Access my data any time
Instagram may access my data when I'm not using the application

Logged in as Ngoan Nguyen · Log Out



Google OAuth 2.0 Playground wants to access your Google Account

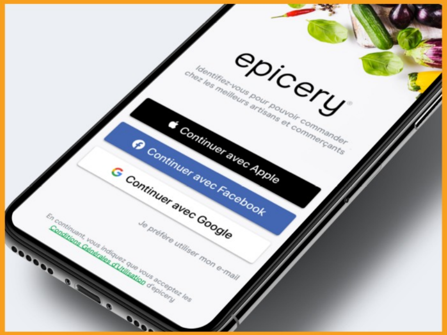
user@domain.com

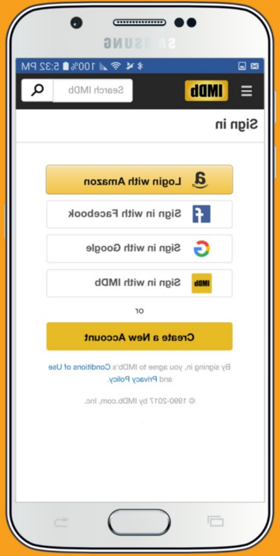
This will allow Google OAuth 2.0 Playground to:

See, edit, create and delete all of your Google Drive files

Make sure that you trust Google OAuth 2.0 Playground

You may be sharing sensitive info with this site or app. Find out how Google OAuth 2.0 Playground will handle your data by reviewing its terms of service and privacy policies. You can always see or remove access in your Google account.


epicery
Identifiez-vous pour pouvoir commander chez les meilleurs artisans et commerçants
Continuer avec Apple
Continuer avec Facebook
Continuer avec Google
Je préfère utiliser mon e-mail

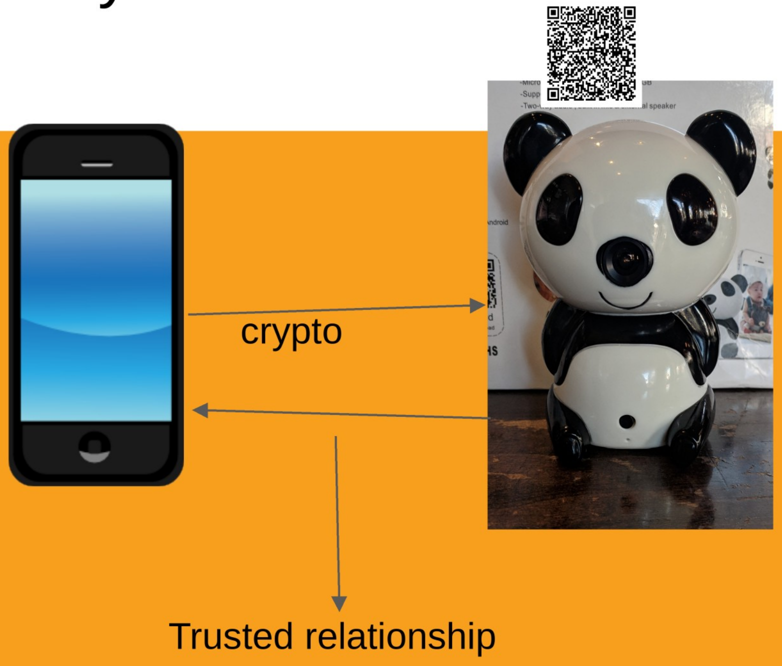


IoT Security Foundation

OXFORD INFORMATION LABS

Self-signed certificates and Raw-Public Keys

- Onboarding process creates initial trust relationship
- Install a self-signed certificate into the IoT thing, private key is in phone
  - Even simpler, just use a Raw Public Key
  - ECDSA keys are small and most M-class processors have acceleration

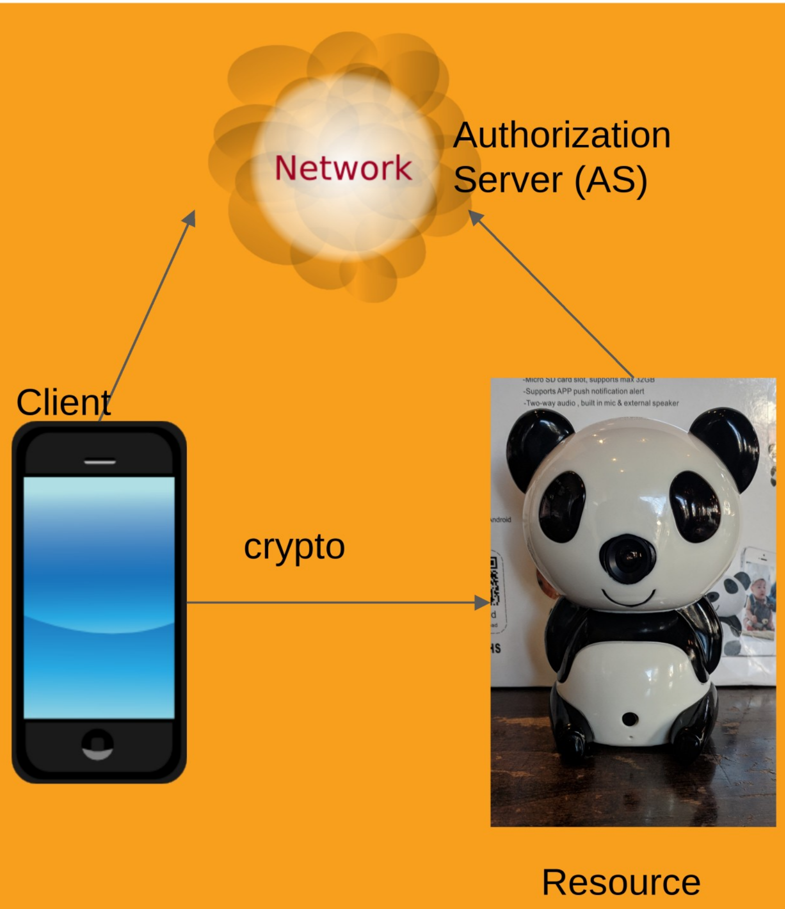


Downside: private key in phone must be kept secure, and available across phone resets, and loss of phone. Can also be hard to share access.

OAUTH2 - authorization

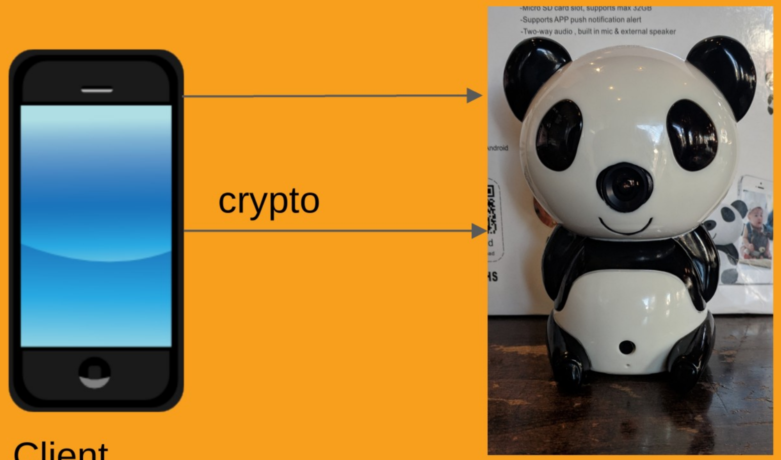
- Device has relationship to cloud.
- Cloud authenticates user using …
- Device is the resource owner
- CLOUD provides authorization token only, not command and control. Client speaks directly to the device.

Downside: cloud retains executive control over device. Device ceases if cloud stops.



Use JWT/CWT directly (use part of OAUTH2)

- Use initial onboarding relationship...
- To create a very-long duration authorization token.
- Device is both resource owner and authorization server.
- Could be used when there are many devices



crypto

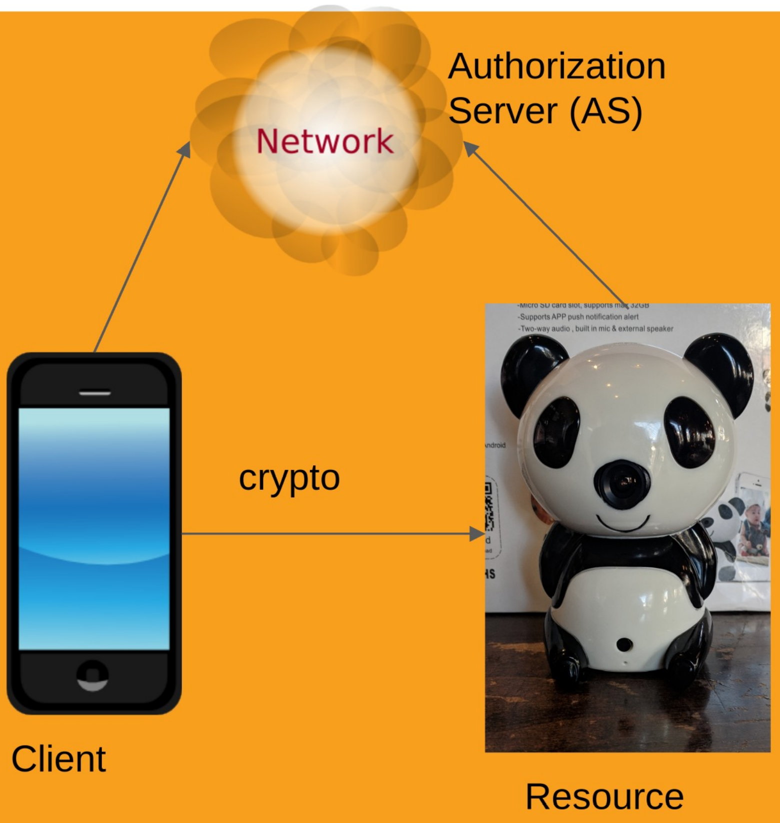
Client

Authorization Server (AS)
+ Resource

CWT/ACE is just OAUTH2, but with JSON->CBOR, HTTP->CoAP, JOSE->COSE.

Magic URL by email

- When user wants access, goes to device, enters email.
- Cloud sends user email with a URL in it.
- URL has token that permits access.
- This can be just OAUTH2, but JSON Web Token is transmitted to authorized user by email.



Upside: cloud can determine authorization by email, user authentication is by email address. Authentication effectively outsourced to Google/Yahoo/etc. Can be effective for certain devices in common areas where only use (not administrate) is desired: treadmill, conference room,

Downside: cloud retains executive control over device. Device ceases if cloud stops. Device can be compromised if email account compromised.

Conclusions and Further Resources

1. Passwords have a long history, and are not loved by users.
2. Default passwords are regularly exploited.
3. Either the initial password has to be changed, or the default has to be very strong.
4. Default passwords related to serial numbers or other public information are easily exploited.
5. Passwords are not always the best choice, and automated onboarding functionality often makes them unnecessary.
6. Cloud integration involves many other trust relationships which can be leveraged to eliminate or reduce dependency upon passwords.
7. Devices seldom care who a person is, but rather, if the person is authorized.

Resources

- Device Provisioning Protocol https://www.wi-fi.org/discover-wi-fi/wi-fi-easy-connect
- Bootstrapping Remote Secure Key Infrastructure (BRSKI)
  - https://datatracker.ietf.org/doc/draft-ietf-anima-bootstrapping-keyinfra/
  - https://openconnectivity.org/developer/specifications/fairhair/
  - Also used in Thread
- Thread Commissioning
  - https://openthread.io/guides/build/commissioning
  - https://www.threadgroup.org/Portals/0/documents/support/CommissioningWhitePaper_658_2.pdf
- Amazon/Google IoT:
  - https://www.trustonic.com/solutions/iot-security/automatic-cloud-enrollment/
- Intel Secure Device Onboarding (SDO)
  - https://software.intel.com/content/www/us/en/develop/tools/secure-device-onboard.html

IoT Security Foundation

OXFORD INFORMATION LABS

Resources

- Android10 + DPP: https://source.android.com/devices/tech/connect/wifi-easy-connect
- https://www.troyhunt.com/everything-you-ever-wanted-to-know/
- https://postmarkapp.com/guides/password-reset-email-best-practices
- https://en.wikipedia.org/wiki/RSA_Factoring_Challenge
- https://en.wikipedia.org/wiki/EFF_DES_cracker
- https://stormpath.com/blog/what-the-heck-is-oauth
- https://www.digitalocean.com/community/tutorials/an-introduction-to-oauth-2
- https://www.supermicro.com/support/BMC_Unique_Password_Guide.pdf

NO UNIVERSAL DEFAULT PASSWORDS

Emily Taylor, Oxford Information Labs

Michael Richardson, Sandelman Software Works

IoT Security Foundation

OXFORD INFORMATION LABS

Course Aims

Participation in this course should result in an understanding of:

- New guidance, EN 303 645, and upcoming regulation
- How users interact with passwords
- Ways to encourage good password hygiene
- How default passwords are causing harm

Learning outcomes

By the end of the course, you should understand:

- The importance of the password user experience
- Technical ways to provision initial passwords
- When passwords are the wrong solution to an authorization problem
- Why password complexity schemes have failed

Security and Technical Education Programme (STEP)

- IoT Security Foundation Quick Guides & training webinars
  - No universal default passwords
  - Managing coordinated vulnerability disclosure
  - Security software updates
- Industry and private sector-led
- Worked closely with UK government
- Technical and regulatory experts

IoT Security Foundation

OXFORD INFORMATION LABS

Who is this course aimed at?

SMEs, start-ups, innovators and researchers

Engage people across the organization…

- Compliance officer
- Product Manager
- Head of Design
- User Experience Manager
- Head of Marketing and Public Relations

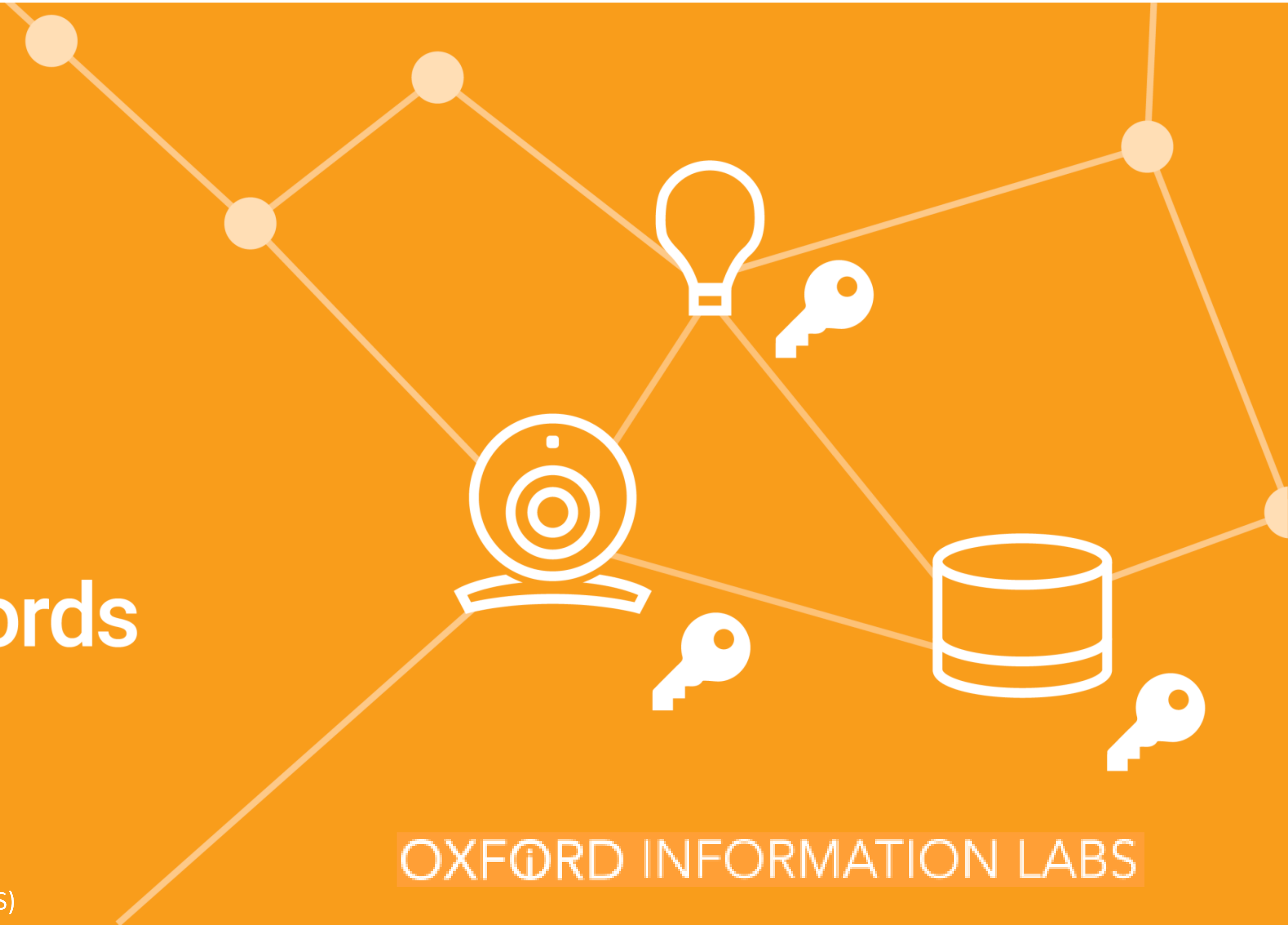
Meet the presenters

Standards and regulation

IoT Security Foundation

OXFORD INFORMATION LABS

What’s the problem with passwords?



60%

of users **don’t** change **device default passwords**

IoT Security Foundation

NETSCOUT Threat Intelligence Report (ATLAS)

OXFORD INFORMATION LABS

Standards and Regulatory Change

Standards

- ETSI EN 303 645 Consumer IoT cybersecurity

Regulation

- US: California Senate Bill #327, Oregon House Bill #2395
- UK: Proposal for regulating consumer smart product cyber security (summer 2020 – consultation on draft legislation)

What guidance is available?

Subject-specific guidance

- IoTSF Quick Guides
- IoTSF Best Practice Guides
- UK NCSC
- US NIST

Codes of Practice

- UK: Code of Practice for Consumer IoT Security
- Australia: Draft code of practice

ETSI: Cybersecurity for Consumer Internet of Things Baseline Requirements

ETSI EN 303 645

- First international standard of its kind
- “Brings together widely considered good practice…baseline provisions.”
- “As consumer IoT products become increasingly secure, it is envisioned that future revisions of the present document will **mandate** provisions that are currently recommendations”

Legislation is making these provisions mandatory

ETSI standard – in brief

Top 3 Covered in this webinar series:

- **No universal default passwords**
- Implement a means to manage reports of vulnerabilities
- Keep software updated

Others:

- Securely store sensitive security parameters
- Communicate securely
- Minimize exposed attack surfaces
- …And more!

UK proposed regulation overview

Aim: Establish a cybersecurity baseline for consumer IoT products

What does it say *now*?

- Applies to network-connectable consumer IoT products
  - “has one or more network interface that can receive and/or transmit digital data”
  - Consumer market, but could be used by businesses
- Sets out obligations for IoT producers and duty of care for distributors
- Products that do not comply should not be ”supplied or made available to consumers” on the UK market

Failure to comply? Fines or removing products from the market

Bans universal default passwords from consumer IoT

Universal default password

- the same password provided with and used in multiple products/devices

Passwords must be…

- Unique per device or set by the user
- Generated (if default) – not derivable or easily guessable

Why?

- Universal passwords weaken security in your product
- Poor password practices put user safety, data, devices, and networks at risk
- Could impact businesses and business continuity

IoT Security Foundation

OXFORD INFORMATION LABS

Look out for other resources in this series

Free! Webinars on Vulnerability Disclosure and Software Updates

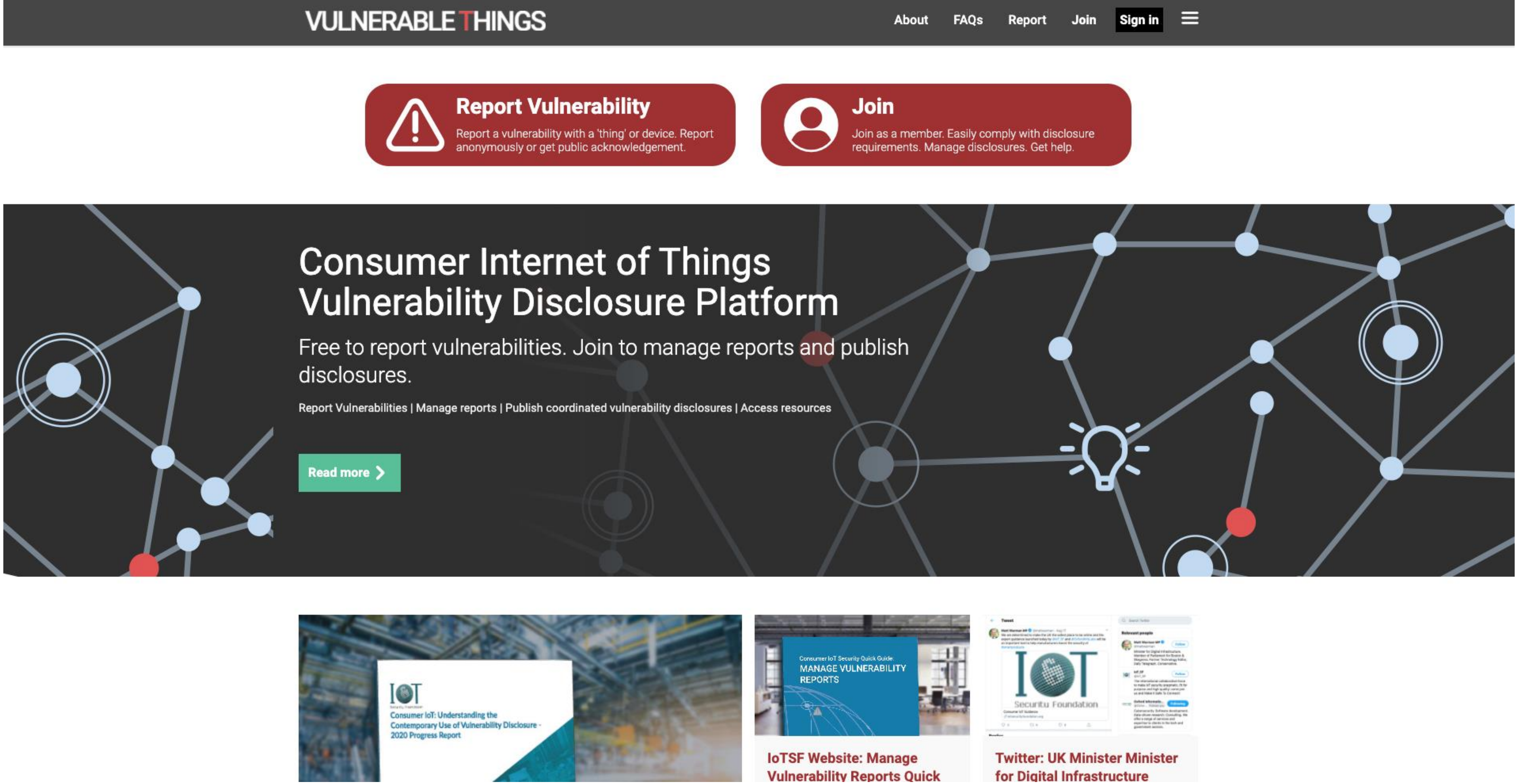
Free! Quick guides to complement the webinar topics https://www.iotsecurityfoundation.org/consumer-iot/

VulnerableThings.com …a vulnerability disclosure platform for consumer IoT supply chain.

Consumer IoT Security Quick Guide: NO UNIVERSAL DEFAULT PASSWORDS

Consumer IoT Security Quick Guide: SOFTWARE UPDATES

Consumer IoT Security Quick Guide: MANAGE VULNERABILITY REPORTS



VULNERABLE THINGS

Report Vulnerability

Report a vulnerability with a 'thing' or device. Report anonymously or get public acknowledgement.

Join

Join as a member. Easily comply with disclosure requirements. Manage disclosures. Get help.

Consumer Internet of Things Vulnerability Disclosure Platform

Free to report vulnerabilities. Join to manage reports and publish disclosures.

Report Vulnerabilities | Manage reports | Publish coordinated vulnerability disclosures | Access resources

Consumer IoT: Understanding the Contemporary Use of Vulnerability Disclosure - 2020 Progress Report

Consumer IoT Security Quick Guide: MANAGE VULNERABILITY REPORTS

IoTSF Website: Manage Vulnerability Reports Quick

Twitter: UK Minister Minister for Digital Infrastructure

IoT Security Foundation

OXFORD INFORMATION LABS