unsecure IoT

KNF-KONGRESS 2025

Roland Wagner

23.11.2025

who am I

Roland Wagner

Auditor

Lead Auditor ISO 27001 (DGI®)

Datenschutz-Auditor (DGI®)

Pen-Testing

OSSTMM OPST Professional Security Tester

OSSTMM CTA Certified Trust Analyst

IT-Forensik

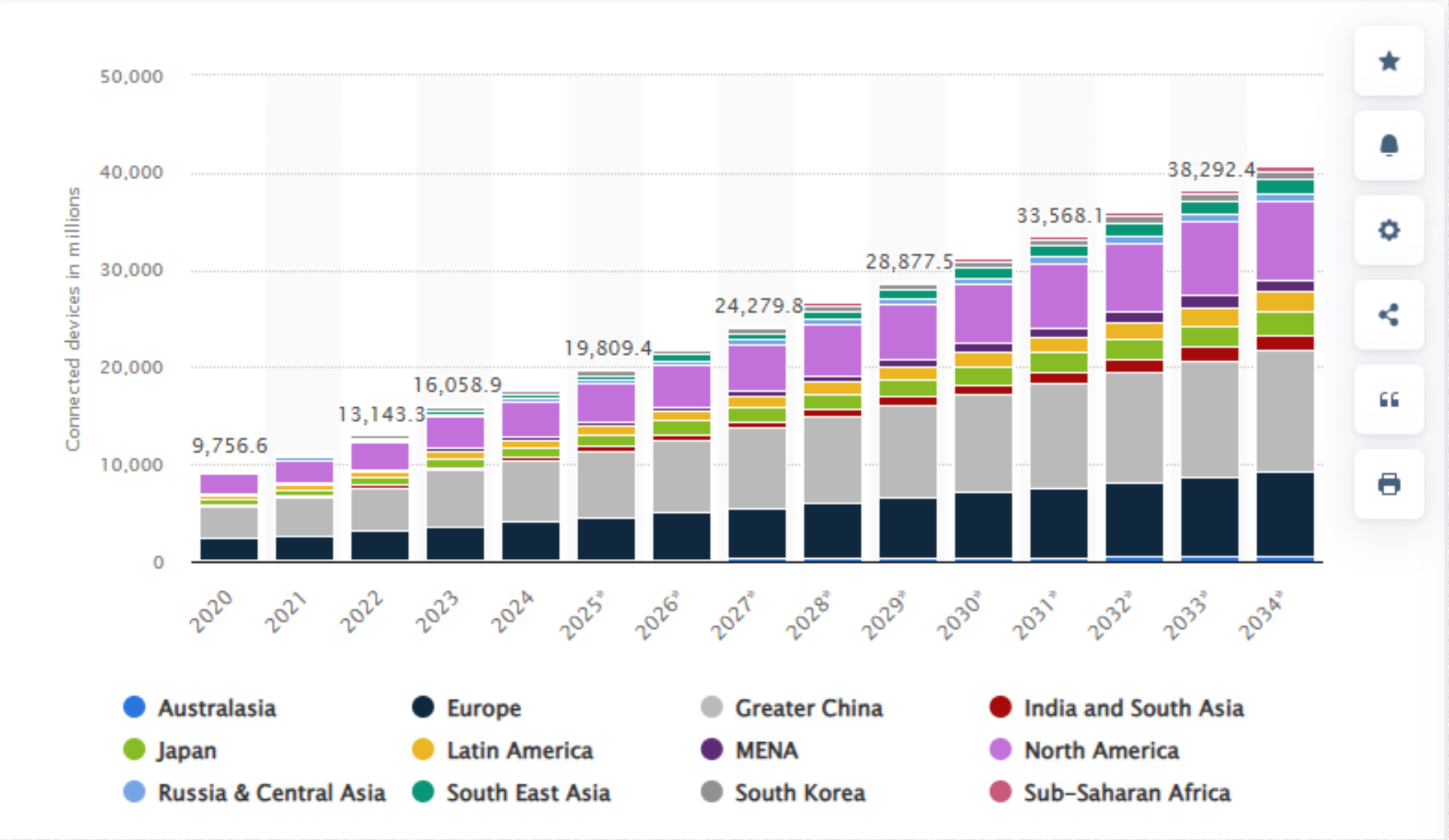
GIAC NETWORK FORENSIC ANALYST (GNFA)

GIAC CERTIFIED FORENSIC EXAMINER (GCFE)

GIAC BATTLEFIELD FORENSICS AND ACQUISITION (GBFA)

IT-Forensik CERT-Spezialist

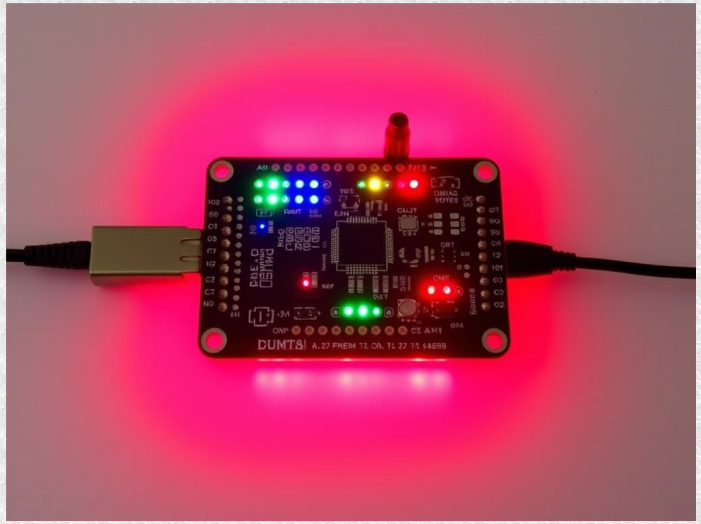
IoT-Devices

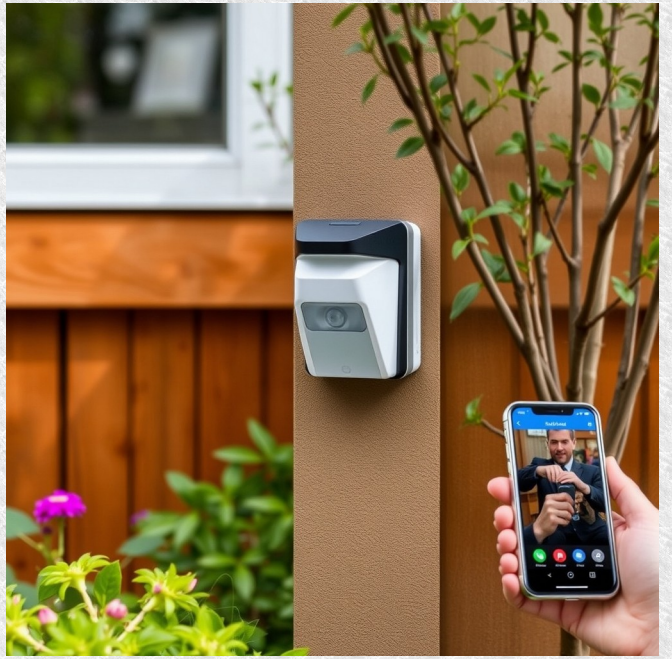


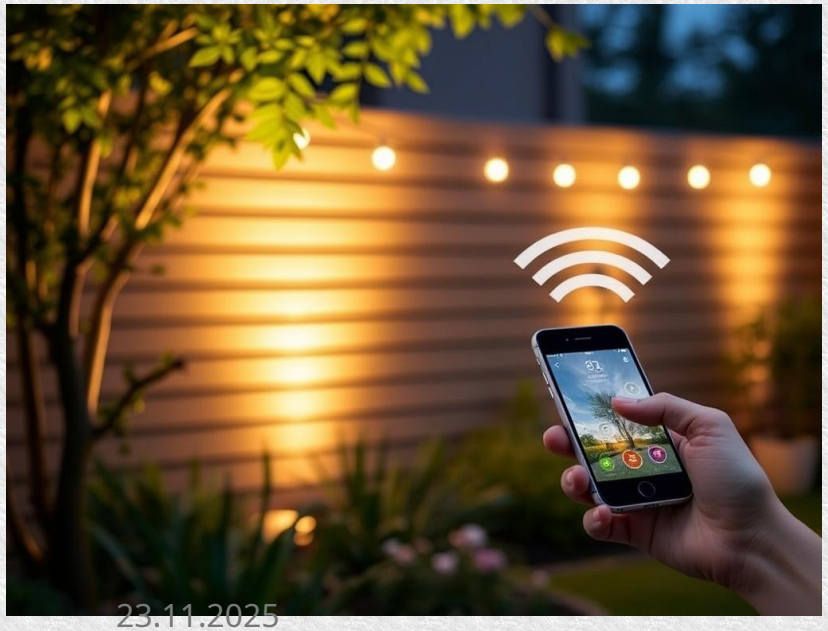
Quelle: IoT connected devices by region 2034| Statista

https://www.youtube.com/watch?v=mnk0KjWxgMA&list=RDmnk0KjWxgMA&start_radio=1

Smart Home







Funktionsprinzip für die Einbindung ins eigene WLAN:

- WLAN-Verbindung ist verschlüsselt (WPA2 / WPA3)
- IoT-Device hat keine Eingabemöglichkeit für Credentials

1. IoT-Device baut einen AP auf (CaptivePortal)
2. Anwender verbindet sich mit AP (default PW für WLAN)
3. Anwender konfiguriert IoT-Geräte als Client im eigenen WLAN
4. Bei erfolgreicher Verbindung zum WLAN wird AP abgeschaltet

Angriffe:

1. „First come“: Konfiguration des IoT-Devices bei der Inbetriebnahme **vor** dem Besitzer

   ABER: RESET durch den Benutzer
2. WLAN mitsniffern:

   ABER: Verbindung verschlüsselt
3. Brute-Force WLAN-Passwort

   ABER: „Viel Glück!“
4. SW-Schwachstelle im IoT-Gerät

   ABER: Zeitnaher Patch
5. Diebstahl des IoT-Gerätes

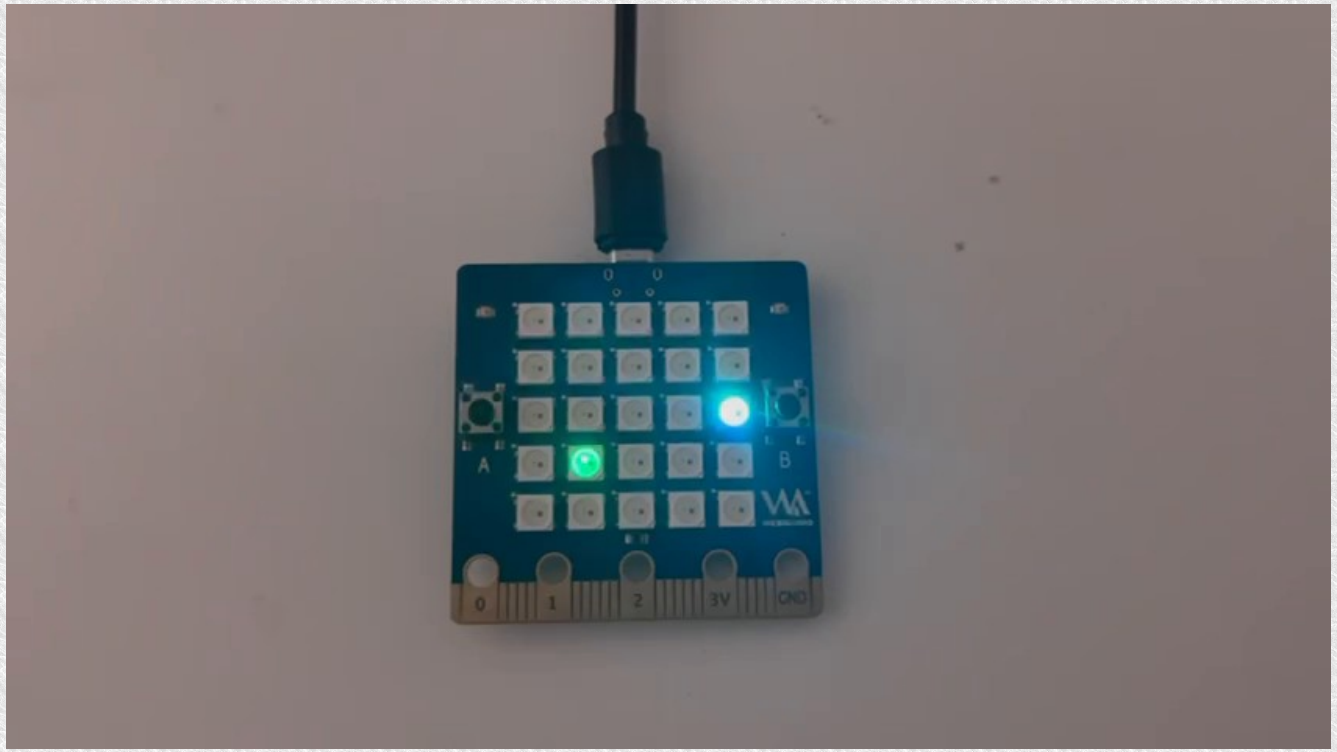
   ABER: Nur bei exponierter Lage

Was nützt mir als Angreifer der Diebstahl eines IoT-Gerätes?

Demo

Was nützt mir als Angreifer der Diebstahl eines IoT-Gerätes?

Demo



Gegenmaßnahmen I:

Diebstahl verhindern

- Mechanische Sicherung
- ➔ verbesserte Sicherheit, aber Aufwand und Diebstahl nur erschwert

Überwachung

- Alarm bei Ausfall (7x24)
- **sofortige** Änderung der WLAN-Credentials an **allen** IoT-Geräten
- ➔ zeitnahe Reaktion notwendig, Aufwand!

Ethernet über LAN

- Kabel exponiert
- ➔ MAC-Security: MAC fälschbar, Überwachung nötig

Gegenmaßnahmen II:

Security by Obscurity

- Kennwort verschlüsseln
- Zertifikate
- Kennwort kodieren (BASE64, BASE56)
- Kennwort nicht im Flash speichern (EEPROM)
- Kennwort gesplittet speichern

➔ Aufwand bei der Entwicklung

➔ verbesserte Sicherheit, aber durch Reverse Engineering dennoch unsicher

Credentials nur im RAM speichern

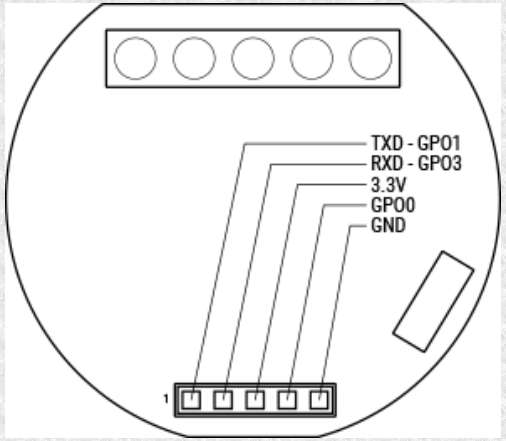
- Zugriff auf RAM beim aktiven IoT-Gerät ? (esptool dump-mem)

➔ Aufwand bei Stromverlust

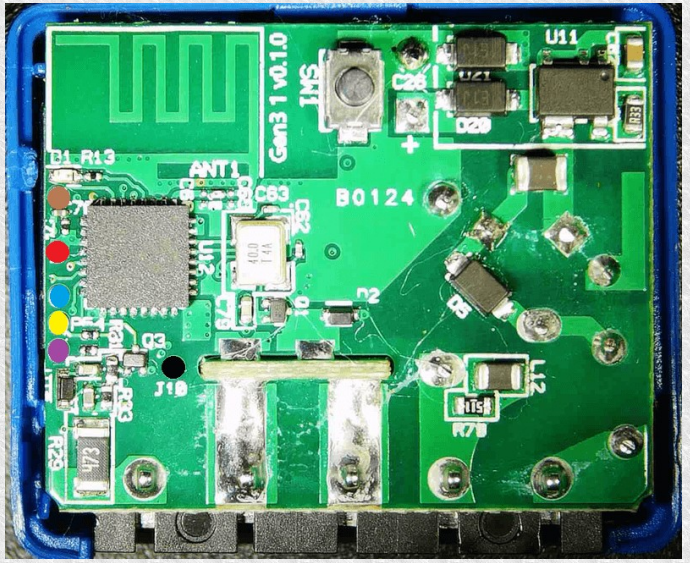
Erinnerung! Voraussetzung:

- (Serieller) Zugang zum IoT-Gerät bzw. zum Microcontroller

- Shelly Gen1



- Shelly Gen3

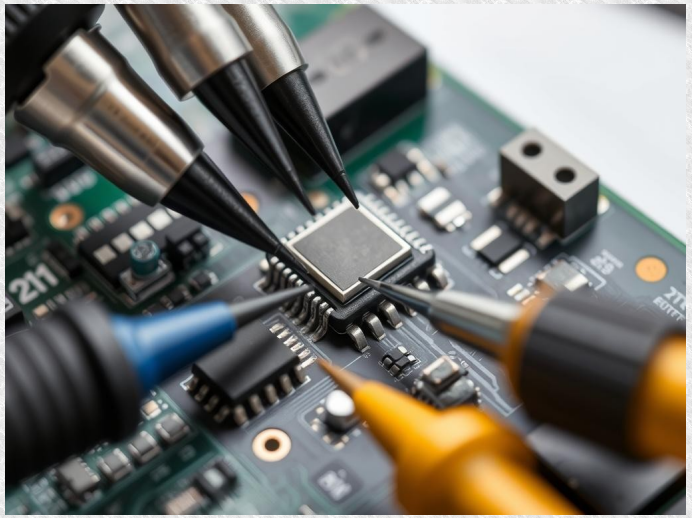


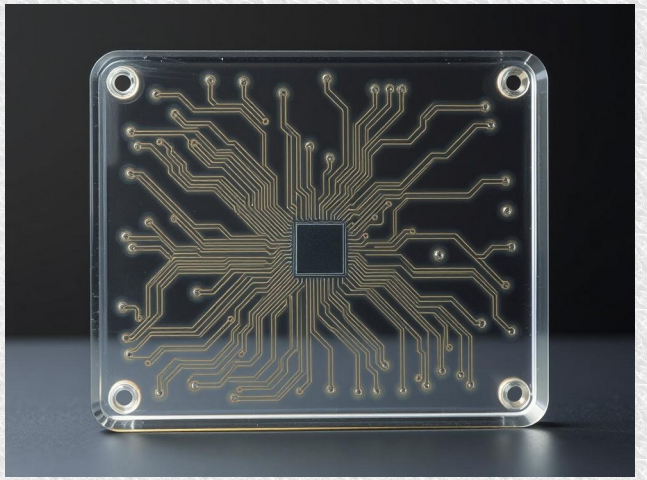
Gegenmaßnahmen III:

- Zugang zu serieller Schnittstelle verhindern
- Keine Testpoints ➔ ToDo: Hersteller

  ➔ Testbed mir Prüfspitzen, Chip-Zange, Chip-Off
- Platine sichern (vergießen in Epoxyd)

  ➔ Aufwand, Kosten, Wartung





Gegenmaßnahmen IV

Zugangsdaten „read-only“ speichern -> „TPM“

TPM für IoT-Geräte ?

- ➔ Aufwand
- ➔ Kosten
- ➔ Breaking Bitlocker

(https://www.youtube.com/watch?v=wTl4vEednkQ)

ToDo:

- Weitere Geräte untersuchen

Fokus:

- Öffnen des Gerätes
- Verwendeter Controller
- Zugriff auf den Speicher
- Sicherungsmaßnahmen (Security by Obscurity) ?

Fragen ???