GSMA Internet of Things

Automotive IoT Security

Countering the most common forms of attack

AUTOMOTIVE IOT SECURITY

COUNTERING THE MOST COMMON FORMS OF ATTACK

THE EVOLVING ATTACKER

Over the past several decades, a pattern has emerged in information security: the attackers are winning, and they are winning faster. Today, there are more tools, information and technology available for breaking into computer systems than ever before. At the same time, the defence of computer systems, which requires constant diligence, resilient hardware architecture and skilled engineers, is often inadequate.

Five years ago, Don A. Bailey of Lab Mouse Security presented the first ever remote car hack at Black Hat Briefings in Las Vegas. Today DEF CON, one of the world's largest hacker conventions, offers a workshop devoted to car hacking that provides hardware tools, free software technologies and canned strategies for bypassing complex security controls.

As interest in hacking grows, not everyone will adhere to the ethical boundaries required of the professional information security researcher. Some individuals will choose to cross the line. Where there are significant weaknesses, criminals will gather to subvert controls in their favour.

Some attackers are employing a new flavour of malware, called “ransomware”, designed to disable a critical system until the victim pays a fee. Many such attacks can cause serious damage.

In December 2015, for example, a three-week power blackout was caused by malware installed at electrical facilities operating the power grid for a small district in Ukraine.

This malware has been active on the Internet since 2007, but was recently updated to subvert controls and damage hardware in industrial control systems. This is the first known power failure intentionally caused by hackers.

As the Internet of Things (IoT) evolves, and industrial systems become better connected, such attacks are likely to increase. Engineers and executives need to ask themselves when, not if, an attack will occur against their IoT solution. The only way to guard against such attacks effectively, and ensure the overall technology is resilient, is by building security into the solution at its inception.

THE ATTACK PATTERN

Attackers tend to target IoT solutions using a conglomeration of methods that stem from the industries and technologies that underpin the IoT.

The IoT is essentially a combination of cloud, network persistence, and embedded technologies that enables physically connected computing systems to provide innovative new services. In other words, the IoT employs existing technologies to enable interactivity and automation.

Thus, attackers can use well-defined strategies and existing tools to seek out vulnerabilities in IoT solutions.

Figure 1 shows some of the components that might comprise an automotive IoT solution.

Figure 1 - Common automotive IoT components and capabilities.

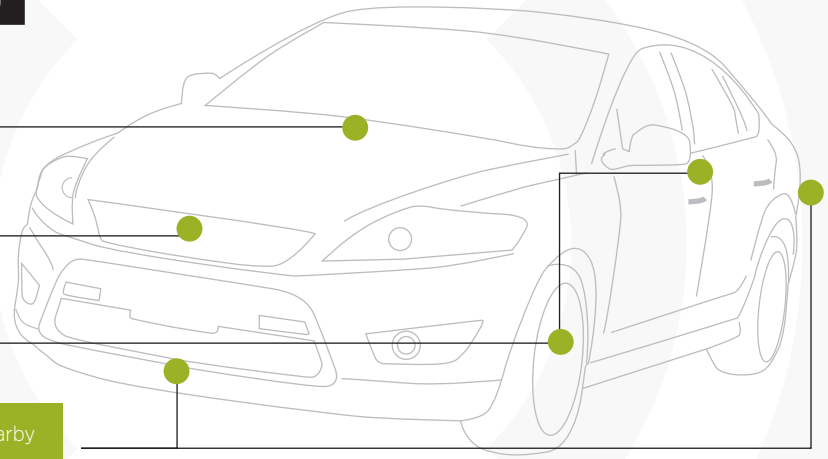
MODERN AUTOMOTIVE IoT CAPABILITIES

Modern telematics systems aggregate data, entertain, and visualize diagnostics

A central computing system guides real-time decision making

Sensors guide drivers toward the safe negotiation of the road conditions

Wireless communication systems interact with nearby peers to relay safety critical metrics and alerts



The common strategies used to attack IoT technologies are:

- Weaknesses in peer authentication
- Practical cryptographic tampering
- Gaps in endpoint integrity
- A lack of segmentation between critical and non-critical applications
- Flaws in software applications
- Business logic weaknesses

Every knowledgeable attacker knows a physical device will be the weakest point of entry into any isolated communications network. Since physical device security is challenging, the easiest way to subvert an IoT ecosystem is by either abusing weaknesses in network communications or weaknesses in the physical endpoint.

Although the core telematics systems might be secured by exceptional engineering, the sensor or ECU (electronic control unit) endpoints that compose the rest of a vehicle's computing network can be difficult to secure because of costs and complexity.

Figure 2 shows some of the ways in which an automotive IoT solution might be attacked.

Figure 2 - Common attack patterns in automotive environments.

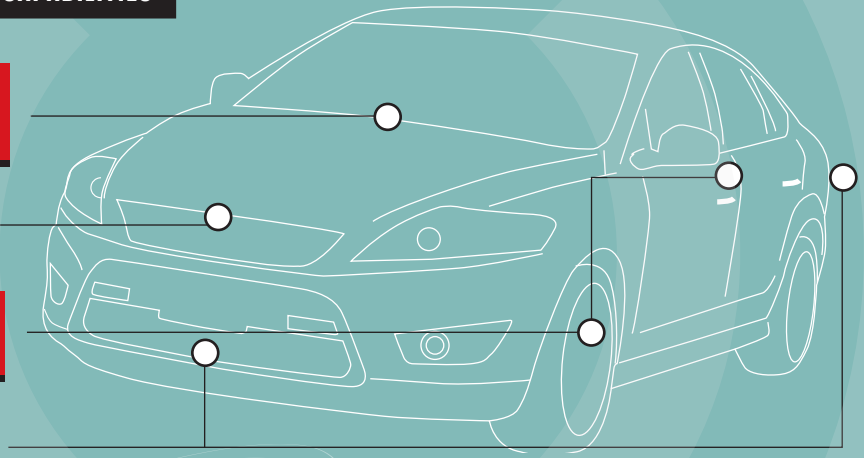
ATTACK PATTERNS AGAINST AUTOMOTIVE IoT CAPABILITIES

Telematics back-end service impersonation, firmware update manipulation, communication security flaws, and third-party application "jailbreaks"

Local or remote CANbus instrumentation to control ECU decision making

Remote code execution or sensor data impersonation via standard wireless protocol weaknesses

Manipulation of critical communication channels by abuse of security certificate or key hierarchies



COST-EFFECTIVE RESOLUTIONS

The issues outlined above are neither systemic nor unsolvable. In fact, there are very cost-effective ways to deter attacks on IoT solutions.

While administrative interface security must largely be addressed separately from the product or service architecture, the following four measures can secure the administrative interfaces made available on the endpoint device.

- **Require the use of a Trusted Computing Base for network and application security**
- **Ensure all network communications are confidential and have integrity**
- **Restrict application behaviour**
- **Enforce tamper resistance**

1. Use a Trusted Computing Base

A Trusted Computing Base (TCB) is a collection of policies, procedures, and technologies that enforce the use and security of critical cryptographic and application-based tokens. It is the foundation upon which a platform's trustworthiness can be defined. If a well-engineered TCB is used at the core of a product, the product will be trustworthy in the field. The use of a TCB can:

- **Diminish or even eliminate the potential for hardware cloning or spoofing**
- **Enforce the use of authentic components within the service**
- **Improve the cost-effectiveness of in-field or remote over-the-air application updates**
- **Increase interoperability and trust between the different components of a service**
- **Improve the longevity of a product**

The GSMA IoT Security Guidelines provide more information on the Trusted Computing Base and can be downloaded from: **http://www.gsma.com/IoT/iot-security-guidelines**

2. Secure Network Communications

The second most important attribute of IoT security is network communications. All components within a network must be able to authenticate one another and, where applicable, communicate data confidentially. These components need to communicate with verifiable integrity, to ensure data cannot be intercepted, altered, or impersonated.

Without a well-engineered TCB, securing network communications can be problematic and often results in unexpected behaviour in production environments. For example, many new IoT products use personal area network (PAN) communications technologies, such as Bluetooth Low Energy (BLE), Zigbee, and Thread.

These protocols include new security features that allow secure sessions to be created between networked peers on an untrusted network.

Although the cryptographic algorithms these updated protocols use (such as Elliptic Curve Diffie-Hellman) to secure a session are mathematically correct, guarantees about data confidentiality and integrity cannot be assured.

That's because these technologies have no root of trust, don't store keys in tamper-resistant areas of memory and may not have certain processing capabilities required for full session security.

Since the first goal of any would-be attacker is the analysis of network communications, it is imperative the security of network communications is considered a critical aspect of any IoT product or service.

3. Restrict Application Behaviour

Application security is exceptionally challenging, even for battle-hardened companies. While core applications designed by a manufacturer's engineering team can be thoroughly audited, modern architectures often allow third-party applications to be loaded on to IoT endpoints. As app stores enable users to access potentially hundreds of thousands of third-party apps, it is almost impossible for all of them to be thoroughly audited.

The correct way to secure applications is by isolating them in jails, virtual machines, containers, or another abstraction that limits both their functionality and their access to critical system devices or resources.

This way, flaws in the software will not result in an attacker breaking out of the application and accessing critical resources, such as the CANbus. In particular, it is crucial to ensure the application:

- **Cannot elevate its privileges to affect the host operating system**
- **Has no ability to gain access to low-level drivers or devices**
- **Cannot influence the behaviour of other critical applications**
- **Has no ability to write to, or read from, the memory or resources of other applications**

Where these rules are enforced, even if an attacker gains code-execution by exploiting a third-party application, or if the application has a 'subtle backdoor', the effects are quantifiable and limited to the compromised application. No other application, subsystem, or host operating system should be affected in any way.

4. Enforce Tamper Resistance

As most IoT attacks are channelled through a physical device, obstructing the analysis of these devices can be a practical way to decrease the likelihood of an attack.

Although a physical device in the hands of an attacker will always be at risk of compromise, physical tamper resistance can be used to complicate the attack process and increase the expense to a point where an attack is no longer practical or cost-effective.

For example, light-sensitive fuses can erase memory if a device's case is opened. Similarly, circuits can be embedded in the device's casing, which disconnect a coin-cell battery and cause critical memory components to be erased, if the device is opened.

Other methodologies are also available to create cost-effective measures that significantly increase the amount of time, expertise and equipment the attacker must use to succeed in reverse engineering or subverting the device's security.

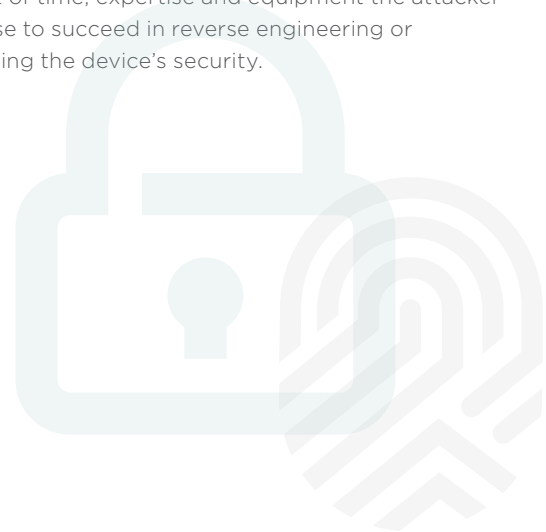


Figure 3 Illustrates some of the strategies that can be used to secure an automotive IoT solution.

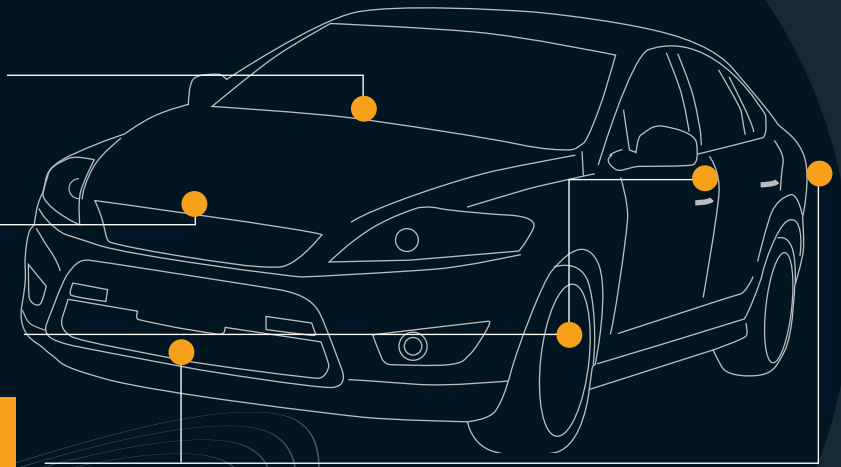
PRACTICAL AUTOMOTIVE IoT SECURITY STRATEGIES

Enforce application level communications security to ensure the highest degree of confidentiality and integrity even when mobile network security is uncertain due to roaming or protocol downgrades

Build trust into the core architecture to decrease long-term engineering cost and improve device longevity

Deploy peer authentication, data confidentiality, and message integrity even on lightweight sensor endpoints

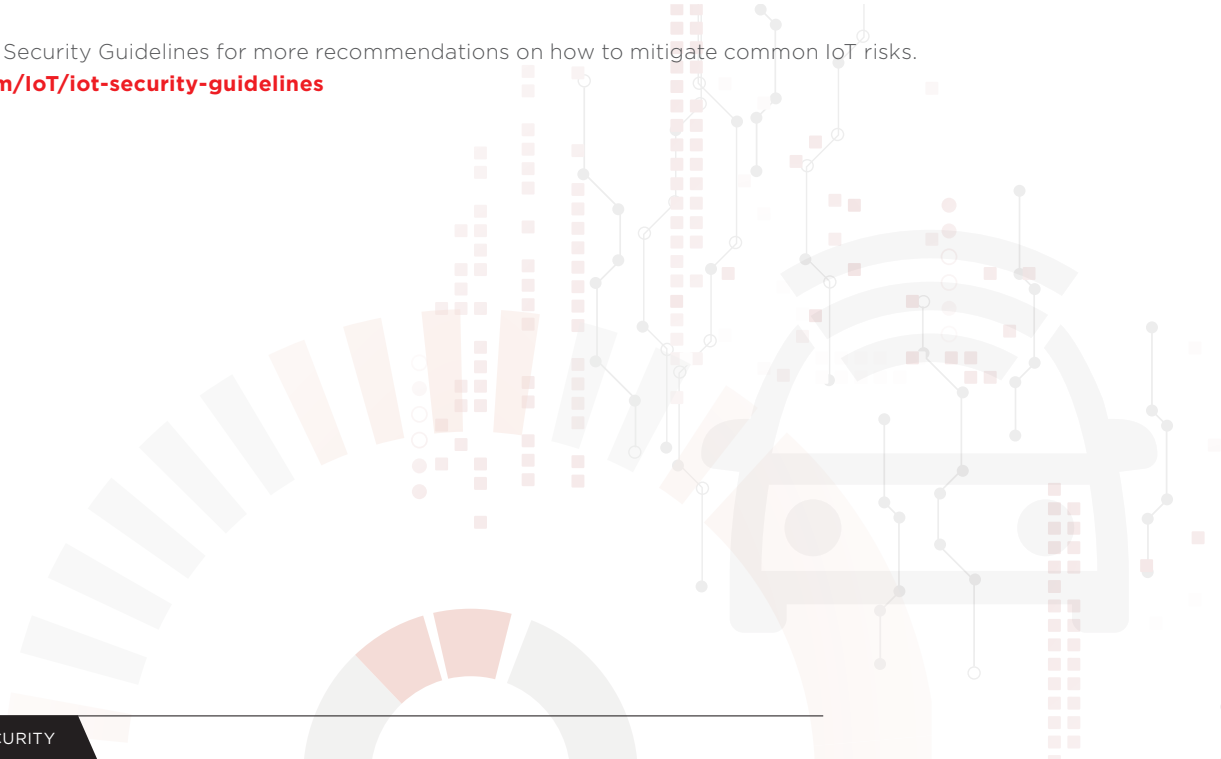
Segment critical and non-critical applications from each other as this diminishes an attacker's ability to affect critical components, if a custom app is compromised



SUMMARY

Despite the media hype, IoT solutions can be secured. Cost-effective security starts at the architectural level. Small changes can ensure the entire IoT product or service ecosystem is safe from abuse. But, in order to achieve this, the engineering team must take the time to build in security from the ground up: Security in IoT solutions cannot be implemented as an add-on. It must be a foundation.

Consult the GSMA IoT Security Guidelines for more recommendations on how to mitigate common IoT risks.
http://www.gsma.com/IoT/iot-security-guidelines



To keep up with all the latest GSMA IoT news:

Visit our website: **www.gsma.com/IoT**
Sign up for our newsletter: **http://www.gsma.com/IoT/sign-up-for-newsletter/**
Follow us on LinkedIn: **http://gsma.at/IoT**