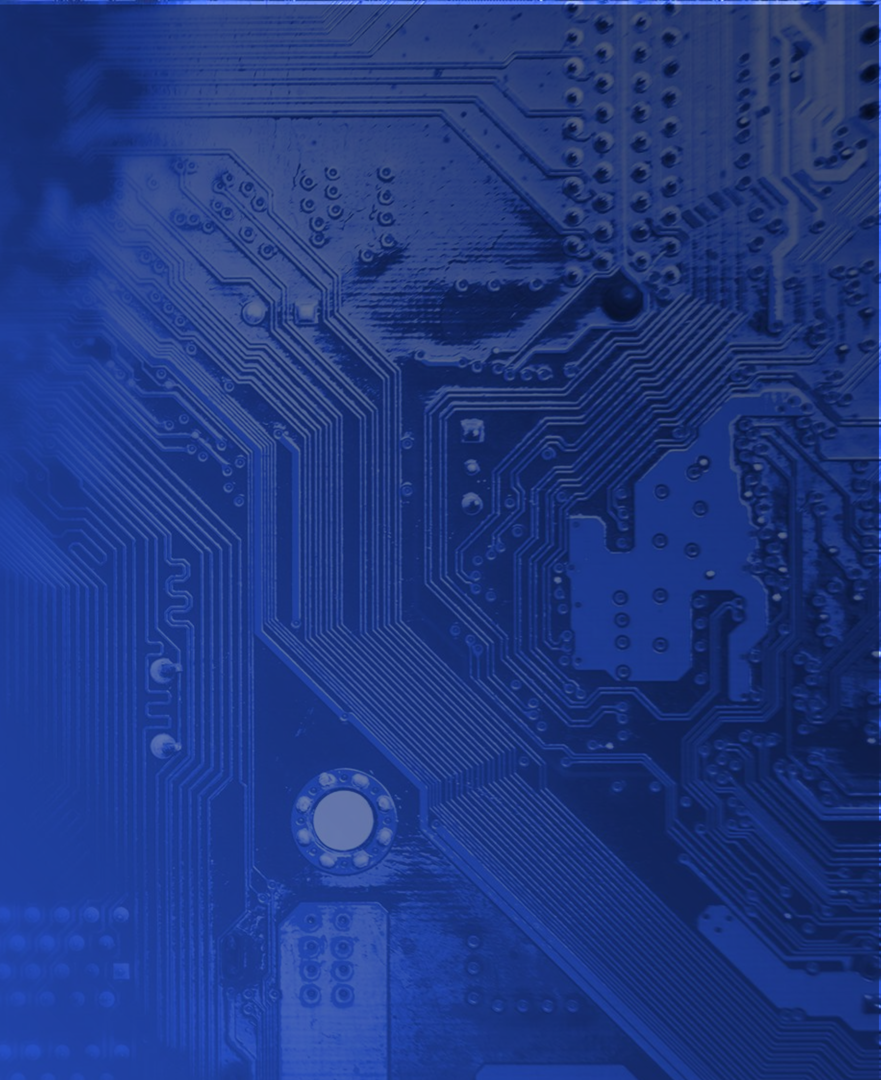


Information Security Services

? How much would it cost you if...

01 Your customers' **payment data** were **breached**?

02 Your apps' **communication protocols** were **insecure**?

03 Your users' **transaction requests** were **intercepted and altered**?

04 Your **platform** was **unavailable for a few days?**

05 Your **network** was **infected with malware or virus**?

06 Your customers **lost the trust in your company?**

07 Your **payment system** was **compromised?**

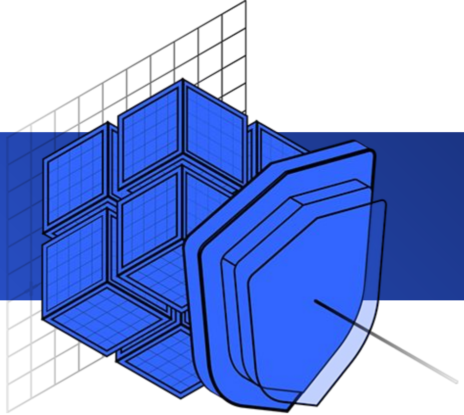
08 Your intellectual properties were **sold to your competitors?**

09 GDPR or CBN were aware of a **successful breach of your data?**

*up to €20M or 4% of annual revenue in possible fines

01

Examples of attacks



Ticketmaster

Third-party code on Ticketmaster's web domain was compromised, leading to the implant of credit card skimming malware on the domain. Up to 40,000 UK and international customers are believed to have been affected.

Under Armour

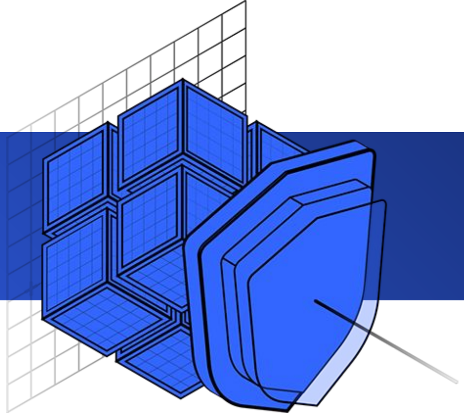
The company revealed that the firm's MyFitnessPal mobile app had been hacked, leading to the compromise of 150 million accounts. Usernames, email addresses, and hashed passwords were stolen.

British Airways

British Airways leaked data belonging to hundreds of thousands of customers who used a credit card to make reward bookings between April and July.

02

Examples of attacks



SingHealth

Singapore suffered the "most serious" data breach in the country's history this year when healthcare institutions group SingHealth's networks were compromised.

Melbourne

A data breach deemed "appalling" affected students at a Melbourne high school, in which their confidential medical and behavioral records were published online.

Facebook

A vulnerability in Facebook's code permitted attackers to steal authentication tokens. Information including names, cities, device types, places of work, and more was also stolen from some users.

WE DEFEND!

One of the primary goals of our cyber security program is to limit the attractiveness for the attacker.

Hacking has moved well beyond the script kiddie threat stage, and the more time it takes an attacker to penetrate a system, the less desirable that target becomes.

About us
Skills

- Computer Network Architecture,
- Network Security,
- Assembly Language (x86 and ARM),
- Machine Language,
- Operating Systems Kernels,
- Virtualization (Intel VT-x,
- Intel VT-d,
- ARM Virtualization),
- Bare-Metal Hypervisor (Type 1),
- Rootkits,
- Binary Hacking,
- Reverse Engineering,
- Threat Intelligence,
- DLP,
- Security Operations Center,
- CICD - SaST / DaST.

Proposed Services

01

Web Application Penetration Testing

02

Mobile Application Penetration Testing

03

Infrastructure Penetration Testing

04

Transaction & Communication Protocol security

05

Application & Firmware Reverse Engineering

06

Continuous Protection and Maintenance

01 Web application penetration testing

Web-based platforms represent a highly tempting target for malicious users, especially when the platform involves payments, monetary transactions and financial management features.

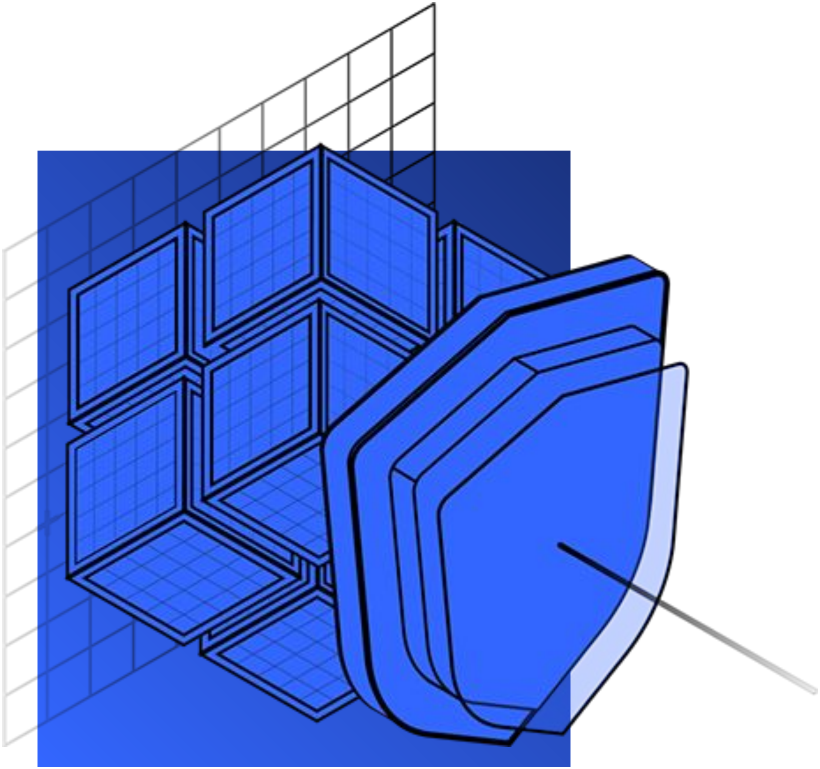
Among the most hunted vulnerabilities that can lead to undesired scenarios we identify:

- File-upload vulnerabilities, path traversal vulnerabilities or improper access management, that can lead to losing control of the web server;
- Vulnerable third-party apps or services, which can compromise the web server;

01

Web application penetration testing

- Cross-site scripting vulnerabilities, which allows an attacker to inject malicious scripts in the user's client browsers to steal user data;
- SQL injection vulnerabilities or improper database backup management, which can lead to the user data breaches;
- Incorrect application logic design, that can be exploited by an attacker to take advantage of the platform's features even though they didn't pay for the service;



02

Mobile application penetration testing

Mobile-based applications provide the same high risk as their web-based counterparts. Nevertheless, the context is different, therefore, the attack surface is different. This is because mobile apps run on different architectures, such as iOS, Android or Windows.

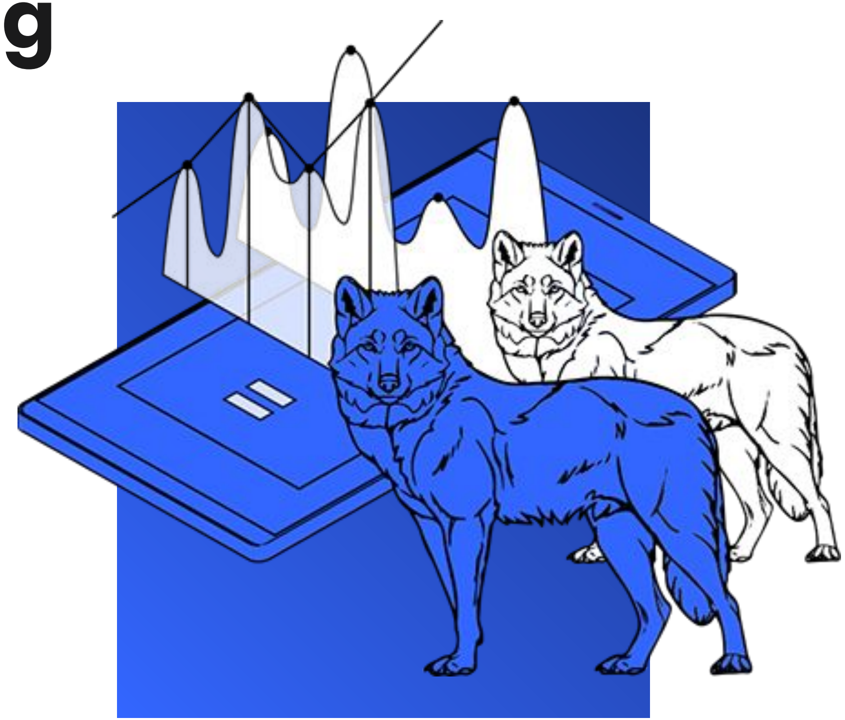
Among the most common vulnerabilities that can lead to undesired scenarios we identify:

- Improper data storage on the mobile device, which can lead to user data breaches;
- Misuse of the permissions of the mobile device, which can lead to user data leakage or compromising the user’s device;

02

Mobile application penetration testing

- Weak or no cryptography when communicating with the application server, which allows an anonymous attacker to misuse the application server or to intercept sensitive user data;
- No robustness against code patching, which allows an attacker that is infiltrated into the user's device to modify the code of the mobile app;
- Incorrect application logic design, which can be exploited by an attacker to take advantage of the platform's features even though they didn't pay for the service;



03

Network Deployment Security

The robustness of a distributed network decreases with every node that is deployed in the network.

A complex topology that includes web-servers, databases, file-storage servers, DNS servers, third-party nodes (such as banks), user endpoints (such as web-browsers or mobile apps) is prone to poor network security misconfigurations because of its size.

Among the most hunted vulnerabilities we identify:

- Improper firewall configuration, which allows an attacker to infiltrate in the network and drop malware or impersonate other registered users or devices
- Missing software and firmware updates, which allows an attacker to exploit publicly known vulnerabilities in third party apps or software

03

Network Deployment Security

- Publicly accessible vulnerable services on the network devices, such as open ports, which may allow an attacker to compromise the device
- DoS and DDoS against the devices deployed in the network
- Incorrect issuing of authentication certificates, that allow malicious devices or users to participate in the network

04

Transaction & Communication Protocol Security

Protocols that transfer sensitive information such as payment transactions or cardholder data are susceptible to attacks because of the value that can be obtained by successfully exploiting their vulnerabilities.

Transaction and Communication protocols tend to misinterpret various input combination, which opens an attack field for malicious users.

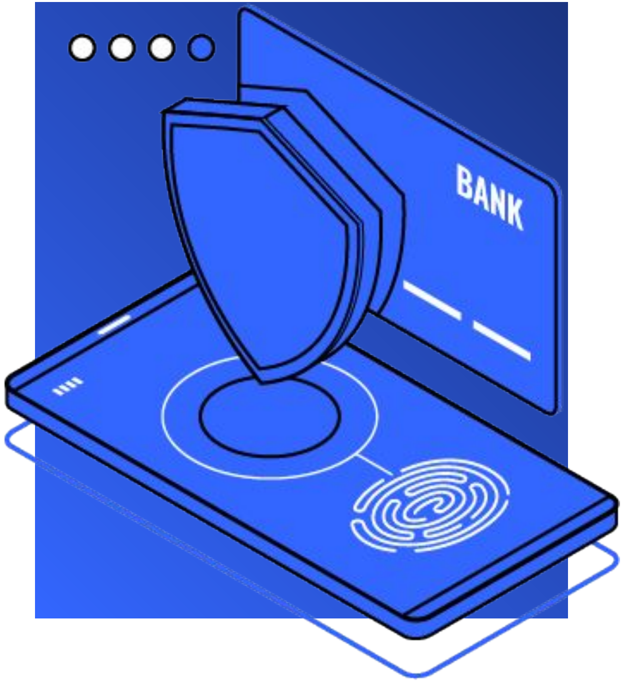
Among the most dangerous vulnerabilities that protocols can suffer we identify:

- Lack or poor confidentiality when transmitting sensitive data, which may lead to user data breaches

04

Transaction & Communication Protocol Security

- Improper integrity protection, which allows an attacker to alter the transactions in their own favor
- Missing input and output sanitization and validation, which may allow a carefully crafted input to exploit the platform or produce malicious outputs



05

App & Firmware Reverse Engineering

When trading proprietary closed-source software or firmware, one must be aware of the following risks:

- The application may present vulnerabilities that can be exploited by a skilled attacker even without possessing its source code
- The application's code might be leaked therefore revealing its implementation and design details
- The application might be patched so that users bypass subscription checks in order to use its features

05

App & Firmware Reverse Engineering

All of these aspects can be tackled by a reverse engineering process. The security analysts that perform reverse engineering can discover before the attacker the vulnerabilities hidden in the application binary.

Therefore, the development team is given time to fix the bugs before the release. Also, via reverse engineering the analysts can assess how difficult it is for an interested competitor to obtain the application design and implementation details.

Last but not least, the analysis team can assess the complexity of bypassing the checks for paid services and can recommend enhancements to the development team for building the application more robust.

06

Continuous Protection and Maintenance

In order to protect the client's systems, platforms and processes against vulnerabilities that target the operating software running on their servers or within their infrastructure, and to remediate quickly in case of a successful attack against these -

-

To foresee further malicious attempts against these, we will deploy the following hardening mechanisms:

- Install anti-virus, anti-malware and IDS solutions on client's nodes in their operating infrastructure
- Enable a logging strategy for critical operations (such as administrator actions, cardholder data access, and others), which can be consulted in case of a successful attack and help recover the client fast

06

Continuous Protection and Maintenance

For keeping the client's services and products up to date with the security standards, and for checking the level of protection on the software installed in their infrastructure, we will perform the following:

- Regularly assess the security of the developed services and products
- Regularly test the installed protection mechanisms on the client's infrastructure
- Regularly audit the logging files, in order to identify malicious attempts

Our process
is easy

01 Analysis

02 Attack

03 Results

Stage 1
- Analysis

Collect Open Source Intelligence (OSINT)

Public information that an attacker can gather about you and your business

Assess tech stack

Technologies used to build your infrastructure

Black-box or White-box or Gray-box?

External blind attack or internal omniscient attack

Present an attack strategy

Steps, timeline, warnings, required permissions

01 **Analysis**

02 **Attack**

03 **Results**

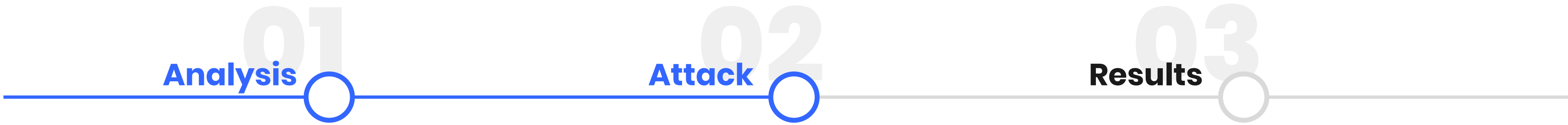
Stage 2
– Attack

Infiltration

- Gain privileged or unprivileged access as an unauthenticated external entity
- Gain privileged access as an unprivileged authenticated user through privilege escalation
- Access privileged resources as an unprivileged authenticated user
- Bypass security checks via misconfigurations in your infrastructure
- Penetrate through the vulnerable public services of your infrastructure

Exploitation

- Simulate malicious actions that an attacker can employ after infiltrating in your infrastructure
- Install malware, leak sensitive information, block services, configure backdoors, and others
- Demonstrate the consequences of the infiltration step



01 Analysis

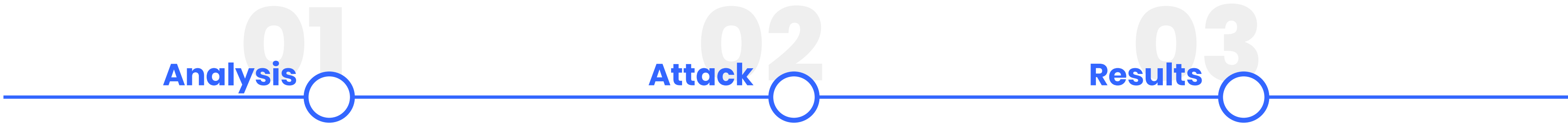
02 Attack

03 Results

Stage 3
– Results

We will compile a comprehensive report that contains the following:

- Executive Summary
- Methodology & Scope
- Conducted Tests
- Vulnerabilities Identified
- List, Distribution, Risk of each Vulnerability
- Severity and probability analysis Risk computing (CVSS)
- Counter-measures and references
- Detailed Report of Each Vulnerability
- Solutions for each vulnerability
- Recommendation



thank you