Misconfiguration and Vulnerabilities

Prevent data breaches from misconfiguration or vulnerabilities of apps and devices

The exploitation of a vulnerability or misconfiguration is the root cause of most data breaches. Most breaches start when a bad actor discovers an opening to the corporate network via a misconfigured device, unpatched software, or default device settings. Around 35% of data breaches can be attributed to a device or application misconfiguration.

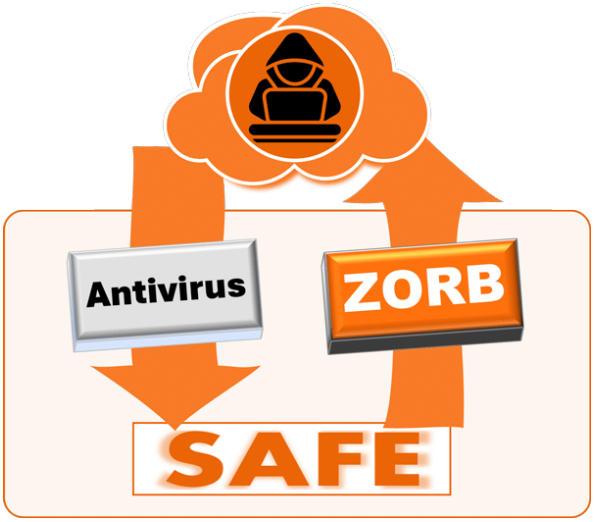
Human error, lack of knowledge, unchanged default settings, or enabling unnecessary features can open a range of doors for an attacker. An estimated 84% of businesses have a high-risk vulnerability on their network perimeter. In 2019, lethargy in patching known vulnerabilities contributed to 60% of data breaches.

Despite regular patching, a determined attacker can force vulnerable code upon a business. MoveIT, Fujitsu, Microsoft are recent high-profile breaches that started with a third parties sending malicious applications updates containing compromised code. Most application updates start with a pull request from the device, with the application querying a central server for updates. Therefore, application update requests are an outbound data flow, that can be redirected.

START TODAY
10 FREE licences
www.zorbsecurity.com/trial

The much-publicised SolarWinds hack in 2020 was a nation state sponsored attack, where the hackers compromised a third party's system to deliver malicious code via a supposed vendor update.

The code included a backdoor granting the attackers access to the SolarWinds Orion Platform. By encoding the malicious code within an update, the code was able to by-pass antivirus and firewall protection.



Prevention

Due to the sheer volume of devices, hardware and applications on today's corporate network, it might not be possible to 100% eliminate risk from vulnerabilities or misconfiguration.

Applying a zero-trust framework to outbound data can eliminate the risk of applications requesting updates from malicious endpoints.

ZORB queries the destination address of all outbound data flows. If the destination address is not tied to the application's vendor IP address range, the data flow is blocked.

Example - Preventing malicious updates

Safelist

Application	Trusted Destinations	Trusted Ports
word.exe	Microsoft	
msteams.exe	Microsoft	
sxdhelper.exe	Microsoft	
waasmedicagent.exe	Microsoft	
sihclient.exe	Microsoft	
Chromeupdater.exe	Google	

A company recently upgraded to Windows 11. However, an old Win 95 PC could not be updated as it runs a core manufacturing sensor. An attacker found a vulnerability on this outdated device and was able to use this to introduce a malware to redirect all Microsoft update requests to a malicious server in China. This malicious update included a backdoor into Microsoft's OS.

The company's safelist has sxdhelper.exe, waasmedicagent.exe, sihclient.exe and other core windows applications as trusted. Additionally these are set as permitted to only communicate with Microsoft. The malware was able to associate itself with the trusted sxdhelper.exe application. However, ZORB noticed that the data transfer was not to a recognised Microsoft IP and blocked the update request.

The company then decided to roll out an AI-based note taking desktop application to all members of its management team. The vendor of the application "ainotes.exe" is www.wedonotes.com

An IT admin logged into their ZORB portal to amend the safelist. They added the application "ainotes.exe" with a trusted vendor destination of "WeDoNotes". They also added "ainotesupdater.exe" with a trusted destination of "WeDoNotes". The amended safelist was automatically updated on all the management team's devices without the user, or IT having to do anything. Data flow, and update requests for this application are now only permitted to be sent to the vendor, and blocked if they are being sent anywhere else.