The Subway Breach

By Cody Scholl

Where and How the Breach Occurred

Breach occurred in roughly 150 Subway restaurants across the United States

Stole 146,000 credit cards in a two years span

This total breach racked up nearly $10 million in losses

Two men found a vulnerable breach in the POS of Subways and stole credit cards

The vulnerability only allowed them to gain access to the desktop but poor passwords allowed them to download malicious software and spying equipment.

Illustration of their Heist Operation

Dolan and Butu send orders and request to the Subway Desktop to overload it until their moment to breach in

Once inside, they download spyware and keyloggers to obtain credit card information and general data

Who's at Fault?

The vulnerability in the software was the reason hundreds of thousands of credit cards were stolen.

The bad part about this vulnerability was that it was public information, which the two men found on the internet which gave them the idea to start the heist.

The software developers should have fixed, or even known about his flaw in their system but it was too late for many citizens.

Human error is also a large factor, with many restaurants having terrible, easy to guess passwords on desktops that allowed them to be so efficient in the rate they set up spyware.

What Failed For the Breach to Occur?

- The vulnerability of their POS system ultimately became the falling domino that lead to all the loss in money and credit cards stolen
- But human error on Subway's behalf also plays a huge importance in this breach.
- Subway had very easy to guess passwords that just allowed these two men to be so efficient in the two years span.
- The software makers of the POS system should be extremely at fault for the failure in their code but also for not fixing or notifying anyone until it was too late.

Preventative Actions

Routine checkups on company computers and constant software updates would help drastically.

Randomized or encrypted passwords would have been a brick wall and possibly have stopped these two from even gaining one computers under their control.

Monthly Scans for keyloggers or spyware would be efficient, they are not easy to find or detect but with some digging you can find them.

Communication between the business and software developers is crucial in finding problems that can cause such serious damage

From a Programmers Perspective

More intensive testing should have been done to prevent something like this.

A stress test or pushing the software to the limits would have revealed this problem before it was released to the public for use.

Looking at how the program allocates data and its ability to keep clear the heap would provide answers to fixing this vulnerability.

An error message or alternate action if an error occurred would have left less of a chance for problems when things are spiraling out of control.

Don't forget to follow GDPR guidelines so your programs can work with Subways in Europe