It's Up To Us To Stop Hackers

Posted At : February 20, 2019 6:20 PM | Posted By : Wil Genovese
Related Categories: ColdFusion, Coldfusion Security, Security

The first month of 2019 has passed and it was full of year end wrap up articles about anything and everything from 2018. Most were fluff articles on pop culture and such. What I found most interesting were the articles that quantified the past year of hacking and security breaches. According to NBC News, **Hackers stole nearly half a billion personal records in 2018.** There were fewer breaches, but the breaches were bigger and worse and more data than ever was stolen. Crypto-miners have improved as well and not in a good way. Previously I wrote about **Cryptojacking** and **Hacking for Bitcoins**. These are malware attacks where hackers install crypto-miners on servers they have compromised. The Crypto-miners use your CPUs to make money for themselves. Hackers have taken this malware to a new level of deviousness. The malware can now target and remove cloud security products as reported **here** and **here.**

It's been a banner year for the hackers. It sure wasn't helpful to us that ColdFusion was at the center of some of these attacks. Last years very unfortunate flaw that allowed a .JSP file to be uploaded to a ColdFusion server due to a missing restriction in the CKEditor that Adobe bundles with ColdFusion. It was a customization by Adobe that caused the issue. Thankfully, Adobe patched the flaw quickly. None the less it had huge ramifications. We heard of a few servers getting breached due to this flaw because the servers were not patched fast enough. It's not just Adobe, but also Apache Struts 2 and Oracle WebLogic that had similar arbitrary file upload flaws.

> Cyber security expert Lorrie Faith Cranor, director of Cylab at Carnegie Mellon University, is troubled, *but not surprised* [emphasis added], by the number of exposed records reported by the ITRC.
>
> "We've always been sloppy when it comes to data security and the hackers are finding creative new ways to exploit that," Cranor said.

We've been sloppy! But! We can be better!
She's right, as a whole we've been sloppy. There's a lot of code that is sloppy and many servers are sloppy messes. And almost no one wants to take time to or pay for the time to clean up the slop. We're all suffering from this technical debt and the bill comes due when a server is breached.

More and more attacks are focusing on the oldest form of hacking, the human element. Some of these attacks are phishing attacks targeting the individual. Others are targeting human flaws in code or server security. It's up to us, the programmers, the developers, the systems administrators, the business managers, executives and owners to stop the hackers. Let's break this down and step it up.

As developers, programmers, sysadmins etc it's up to us to stay up to date on security news and security best practices. We need to look at everything we do from a security first point of view instead of an afterthought if a thought at all. This is an easy mantra, but what does it mean? It means using best practices that look at security as part of the solution. These are some best practices for coding web applications.

- **Validating Input:** Validate input from all untrusted data sources such as forms and urls. Proper input validation can eliminate the vast majority of web application vulnerabilities.
- **Sanitize data sent to other systems:** Systems such as database or file systems. With ColdFusion this is using cfqueryparam for all queries! No excuses here! For file uploads, using CFFILE, make sure the files are what you are expecting, placed on the filesystem somewhere safe with no permissions, and are files that cannot be executed.
- **Default Deny:** By default users are denied everything. Only add permissions as needed.
- **Adhere to the principle of least privilege:** This applied to things like files that need to be uploaded, do they need local system or root permissions? NO! This also applies to things like ColdFusion's datasources. Does the datasource need SA access in the database? NO! Datasources should be setup with user accounts that have limited privileges.
- **Defense Layers:** Think of overall application security like layers of an onion. If an attack makes it past the outer layer, the next layer may actually stop the attack. ie, validating form and url variables is the first layer. Using CFQueryParam is the next layer in taking user input and inserting it into the database. Ensuring proper datatypes in the database would be the next layer. Each layer works together to prevent attacks.
- **Keep it simple:** Keep the web application design as simple and small as possible. Complex designs increase the likelihood that errors will be made in their implementation, configuration, and use. Additionally, the effort required to achieve an appropriate level of assurance increases dramatically as security mechanisms become more complex
- **QA:** I've heard the phrase "We refer to our users as 'The QA Team'. It's funny, but ineffective and irresponsible. Your team should utilize code audits, peer review, pen-testing, fuzz testing, etc. All of these techniques helps ensure higher code quality and higher levels of security.
- **Adopt a secure coding standard:** Develop and/or apply a secure coding standard for your target development language and platform.

The development team as a whole is responsible for quality code. No one should be afraid of peer or code review. We've all found our share of "what the heck" in our old code and the code of others just as others have come to us saying "what the heck?". It happens. Own it, fix it, and learn from it.

As an IT team, including systems architects, systems administrators, it's up to us to make sure the systems the code and data live on are safe and secure. Servers should be maintained by systems administrators not developers. Here are some best practices for administering servers, with special focus on ColdFusion. Some of these are the same concepts as above, but applied differently.

- **Document Everything:** Document each and every server you install and configure.

- **Automate or script tasks:** Common tasks especially common complex tasks should be scripted and even automated. This helps prevent human mistakes.
- **Default Deny:** By default users are denied everything. Only add permissions as needed.
- **Defense Layers:** Think of overall application security like layers of an onion. If an attack makes it past the outer layer, the next layer may actually stop the attack.
- **Adhere to the principle of least privilege:** Do not grant root or local system access to anything and everything. For example ColdFusion does not nor should not be installed as root or local system. The same goes for the database service.
- **Minimize server purposes:** Or Keep It Simple. However you look at it, you should split up the roles of your application across multiple servers. ie. put the database on it's own server separate from the web server. If more servers are needed such as file servers, etc, then separate them. If one gets compromised it should not allow all the others to be compromised.
- **Document and Follow Security Protocols:** This goes for everything from installing a new server to adding new users. Document the protocols for security and follow them every time.
- **Monitoring:** Monitor your systems for up time, down time, CPU utilization, disk utilization, memory utilization, etc. The fastest way to know if a server has been breached is to see a change in its expected behavior. Crypto-miners are notorious for running at 100% CPU utilization.
- **Backup everything:** Do I need to say more? if it exists, back it up.
- **Test your backups:** Backups are great unless they do not work. Test your backups and recovery process from beginning to end and document that process.

The IT team should encourage documentation, lots of verbose documentation. Everyone should be able to reproduce everyone else tasks from the documentation and know that the tasks are secure. How many of us can truly remember what we did last week or last year or longer ago?

Can I take a moment to talk about passwords? The password rules we've been taught for years are wrong. Don't take my word for it take the word of smarter people than me. Go read **this** and come back. The passwords we're taught that look like this Ti%tj89k+ are easy for computers to guess and hard for us to remember. It's easier for us to remember groups of words and connect them together with numbers and special characters such as Star*Night*Cold*Beer45 or some other such nonsense. It's easy for humans to remember, it passes most inane password validation rules, but much harder for computers to guess due to the length of the password. (note to self to never use that as a password). Use **this to generate passwords**.

For the management team I have these words of advice. "Just secure it!" Many decisions in business are based cost benefit analysis. Please review the costs of the known data breaches. The jobs lost, the lawsuits filed, the fines levied, and in rare cases jail terms served. Then ask yourself which is cheaper, the minimal effort and costs upfront to have your development and IT team ensure security or the fallout from having been breached? Even if the breach is nothing more than a "*public relations*" nightmare, can your business survive that?

CF Webtools Developer Teams are ColdFusion experts and are ready to build your applications. We are also an Amazon Partner. Our Operations Group can build, manage, and maintain your AWS services including ColdFusion servers. We also handle migration of physical servers into AWS Cloud services. If you are looking for

professional AWS management our operations group is standing by 24/7 - give us a call at 402-408-3733, or send a note to operations at **CF Webtools** .