Innovation on Trust

Once upon a time, in order to secure a computer, an organization only had to worry about physical security. After all, computers took up an entire room and were programmed using punch cards that were fed into them. However, as computer technology progressed, we began to network them and there were some early issues, such as a hack described in *The Cuckoo's Egg* by Clifford Stoll.[1] At approximately the same time, "personal" computers (PCs) were becoming more ubiquitous, and we then had to worry about the distribution of malware using media. The first few examples were harmless. Usually, they played a prank like flipping the person's screen or changing the colors or making weird sounds come out of the horrendously bad PC speakers.[2]

But as systems have gotten more powerful, as we have networked them more and more, as hacking data has become more profitable, we have had to get smarter and better with regard to computer security. First, adversaries primarily went after weaknesses in applications, default configuration or misconfigured systems. However, as we got better at hardening systems, attackers have focused on people. At some point, we have to trust people. But we have to verify they are who we think they are. This has led to regular innovations around trust and identity. So let us walk through some of these innovations and build up to where we are today.

Stronger Passwords

As more attacks focused on identity, we started looking at how to make passwords harder to guess. Part of it was the technology. For instance, we increased the length of passwords to prevent against brute-force attacks. It is exponentially easier to crack a four-character password vs. an eight-character password. It became computationally feasible to brute-force shorter passwords.

We then added complexity requirements. After all, if we could force someone trying to crack a password to have to consider more possible values for each character slot, more attempts would be required to brute-force a password.

However, attackers started determining that users were reusing passwords, basically resetting passwords until they could get back to the one they had before, so we started keeping password history and putting in minimum amounts of time before a user could change a password again. Today, we usually see 24 remembered passwords and a minimum of a day between changes or something in those ranges. The user can roll back to the original password, but it is going to take 25 days to do so. That is usually not worth their time.

We also started looking for known words in passwords because that proved effective. For instance, in a penetration test in the US State of

K. Brian Kelley, CISA, CSPO, MCSE, Security+
Is an author and columnist focusing primarily on Microsoft SQL Server and Windows security. He currently serves as a data architect and an independent infrastructure/security architect concentrating on Active Directory, SQL Server and Windows Server. He has served in a myriad of other positions including senior database administrator, data warehouse architect, web developer, incident response team lead and project manager. Kelley has spoken at 24 Hours of PASS, IT/Dev Connections, SQLConnections, the TechnoSecurity and Forensics Investigation Conference, the IT GRC Forum, SyntaxCon, and at various SQL Saturdays, Code Camps and user groups.

South Carolina, I watched a pen tester try one particular password against every user account in the organization: **Clemson1**. If you look at that password, it meets most complexity requirements as it has an uppercase and lowercase letter as well as a number. It meets the standard for most industries at the time because it was eight characters in length. Unsurprisingly, that one password worked for about 1 percent of the users. When you have a user base of several thousand users, that means a number of accounts with an easily guessed password in the double digits. Why that particular combination? South Carolina has two major schools: the University of South Carolina (Columbia, USA) and Clemson University (Clemson, South Carolina, USA). And with the natural in-state rivalry, fans of both schools like to proclaim their school as number one.

Early Multifactor Authentication

However, simply increasing the password length and complexity did not prove enough. There are plenty of techniques to grab a password or a password hash and then use those to compromise systems further. Modern operating systems have a number of protections against these sorts of attacks, but new ways to carry out these types of compromises are still being developed. Go back about 10-15 years and the attacks were even more successful. Go back farther and there were little, if any, protections at all.

As a result, we realized that we needed more than just username/password to secure resources. We started using cards such as the Fortezza Crypto Card,[3] which helped establish identity. It was an example of what you know (password/pin) and what you have (the card) and, thus, an example of multifactor authentication (MFA). We also used systems that could calculate seemingly random numbers based on a hidden algorithm. The system could know what the number was and the user could as well because of a hardware device such as a token, which kept displaying the current number. If you got either the password or the number wrong, you did not get in. That algorithm is the key, which is why, in 2011, attackers went after RSA's SecureID algorithm and targeted RSA's systems, eventually leading to a breach.[4]

Password Managers/Vaults

However, MFA did not have widespread usage at that time. It was still used primarily for sensitive enterprise or government resources. As a result, we went back to making passwords longer and increasing complexity requirements. We also evangelized the use of different, hard-to-guess passwords for each site. Corporately, we aggressively tried to apply the principle of least privilege and limited account re-use, especially service account re-use as much as possible.

> "IF IT FEELS LIKE WE ARE CONSTANTLY IN A RACE AGAINST ATTACKS ON IDENTITY, THAT IS BECAUSE WE ARE, UNFORTUNATELY."

However, if a user is asked to use passwords that are hard to remember, that means they may be hard to remember. Therefore, password manager technologies started being developed and championed. The best use a central encryption key that can only be decrypted by the user using a secret they know, like a master password. Corporately, we also use the term "password vault," because we know the passwords must be protected. And, since the proliferation of systems have meant a lot more service accounts to keep track of, and typically the password change interval is less than for normal users, those in charge of rotating the passwords do not remember them (nor would we want them to do so). That has led to enterprise-level technologies that include the ability to implement access controls and even automate password rotation of systems that support such functionality.

The Increase in MFA Usage

And that leads us back to MFA. If it feels like we are constantly in a race against attacks on identity, that is because we are, unfortunately. That leads to the

question, "If passwords are now complex and computationally not worth it (or impossible) to crack, but I still want access, what can I do?" And our adversaries came up with relatively simple solutions that attack our trust. Phishing attacks, which ask you to click on a link to check the status of an order or a discrepancy with your bank account have continued to work. In some cases, attackers have created complete mock-ups of the login experience and are able to grab the password. Some are even sophisticated enough to send the user through to the real site, which preserves the deception a bit longer.

> "SYSTEMS WE WOULD NOT HAVE BOTHERED SECURING WITH MFA A DECADE AGO NOW HAVE SUCH FEATURES REGULARLY."

And if it is not a mocked-up site, it could be the deployment of malware with a keystroke logger. Some of these are sophisticated enough to take screenshots to know what is being sent through and if there is anything else that needs to be done, such as identifying a picture.

Knowing these attacks exist, how have we countered? Typically, we go with a more expansive use of MFA. For instance, most social media platforms have the capability for MFA. An example is the Facebook Code Generator, which uses the application (app) on a mobile device to display a code, which a user can use to log on to the site on a new device. However, many systems still rely on email and text. Some may also offer voice with a text-to-speech system that calls the user and says the code verbally rather than in a message.

Systems we would not have bothered securing with MFA a decade ago now have such features regularly. Today, it is a shock to me when I deal with a vendor that has a username/password system without MFA. They exist, even in some major offerings, but those cases are becoming increasingly rare.

Push Notification and Other Technologies

However, what if the adversary compromises the format by which you use MFA, such as email or text? We know short message service (SMS) is now considered insecure because of weaknesses in older protocols[5] or other mechanisms of attack. Or what about cases in which users do not want to bother checking their email or receiving a text message to have additional protection?

Those issues have led to other mechanisms such as push technologies, where one gets notified of a login attempt via an app on a mobile device and confirms their identity. Such push technologies are extremely handy. They pop up a message and the user can click yes or no.

There are also other solutions such as the website displaying a bar code or Quick Response (QR) code that is read by an app on the mobile device, and that is the input the app uses to generate a code that allows the user to log in. This might be done only the first time a user logs in and it is effectively the same as traditional multifactor solutions with an algorithm. It is the bar code or QR code that transmits the information to synchronize the app with the system rather than a serial number or the like that has to be entered manually.

The Future

Where do we go from here? The problem of authenticating a person's identity and protecting that authentication and identity from attack is not going to go away. We should expect a continued race between those who protect and those who attack. It is difficult to say what is next.

"Passwordless technology" is a new buzz phrase, but the methods behind it are not. MFA solutions have comprised a number of paths (e.g., biometrics, algorithms that generate apparently random numbers, push notification to a device, the use of a hardware device) and basically use more than one method, with the exception of passwords. It is likely that we will see more developments along these methods to try and move us away from passwords. However, we should also

Enjoying this article?

- Read *Audit Oversight for Onboarding Vendors. www.isaca.org/audit-oversight-foronboarding-vendors*
- Learn more about, discuss and collaborate on audit and assurance in ISACA's Online Forums. *https://engage.isaca.org/onlineforums*

fully expect that adversaries will develop counters to those methods.

Endnotes

1 Stoll, C.; *The Cuckoo's Egg: Tracking a Spy Through a Maze of Computer Espionage*, Doubleday, USA, 1989
2 Lewis University, "The Evolution of the Computer Virus," Romeoville, Illinois, USA, *https://online.lewisu.edu/mscs/resources/the-evolution-of-the-computer-virus*
3 Crypto Museum, Fortezza Crypto Card, *https://www.cryptomuseum.com/crypto/usa/fortezza/index.htm*
4 Jarmoc, J.; "RSA Compromise: Impacts of SecureID," Secureworks, 7 March 2011, *https://www.secureworks.com/research/rsacompromise*
5 Whittaker, Z.; "Two-Factor Security Is So Broken, Now Hackers Can Drain Bank Accounts," *ZDNet*, 4 May 2017, *https://www.zdnet.com/article/two-factor-security-is-so-broken-criminals-drained-a-persons-bank-account/*