**RedZone Podcast Episode #82: Passwords Redefined | Make Your Password Interesting, Not Difficult! – with James Crifasi, VP & CTO, RedZone Technologies**

Bill:

[00:00:30] Let's talk about passwords. We have James Crifasi here today, who's the CTO of RedZone. We had a couple of interesting happenings this past week that we thought would be good to share with the audience. One of them is passwords, the creation of adequate passwords for environment. Do you think the old ways of creating passwords potentially might be done?

James: I mean, probably. We've been called in for quite a few issues that were classified as hacks, where in reality someone lost their password. If we look at-

Bill: Lost or got hacked?

[00:01:00]  
James:

[00:01:30]

[00:02:00]  
Lost their password or someone guessed their password. No indications of someone actually doing the effort of hacking. For all we know they wrote it down on a Posty and lost it. If we look at that, then recently we had a security incident where we were allowed to take about 1,500 different accounts and crack them to answer the question of, "How good are the passwords?" We took these 1,500 accounts and what we were able to find is that most of what IT trains users on, and kind of the IT passwords themselves that were thought to be strong, were cracked very, very quickly. This became important because a lot of the calls that we get about hacks in reality are really just someone logging in as the user, using their password. If we take this as an example, out of 1,500 accounts we were surprised to figure out that on a really, really low class slow machine that was not good at all, about 250 of the 1,500 were cracked instantly. That means-

Bill: 250?

James:

[00:02:30] 250, that's microseconds. Not fast, but instantly. Past that if we look at passwords that were cracked in within 20 seconds, it was almost about 600 of the 1,500 could be cracked in 20 seconds. Almost 900 of them could be cracked in five hours. If we look of this in terms of a malware. That means a simple malware running on a relatively low class end user PC can crack the majority of passwords in an environment, and no one will even notice that the crackers running, because it doesn't need much CPU or RAM and no one has to exfiltrate data. What we also found is in most environments the complexity requirements were on. A lot of the passwords were 11-15 digits long.

Bill: The complexity requirements are on?

[00:03:00]  
James:

[00:03:30]  
Are on, that's the, you need to have three of the four upper case, lower case, number, special symbol. Your standard pieces there. They were also 11-15 digits. That's a pretty long password. It was fun from the IT point of view to run the cracker, because we found that we had some five digit passwords that actually outlasted the majority of the 10-15 character passwords. What we realized is that how the crackers are working is a little bit different than what everyone's been trained and what we've really looked at. Just as an example, a lot of people have been trained when you create a long word password you sit there and you say, well E I can replace with a 3, and A, I'll replace with an @ sign, and I, I'll replace with an !. People have learned the mnemonics on the keyboard to figure out, "Okay, how do I remember all these annoying special characters and still have password that I don't have to write down all the time?"

[00:04:00]

[00:04:30]

[00:05:00] What we've found is anyone doing that, basically with the amount of research going into spell checkers, and predictive typing, and things like that. All of that's being built into the dictionary for the cracker. All of those common replacements are actually built into the system. We had some very, very long complicated passwords that were found almost instantly because they were a standard replacement of characters. We had one that as an example, had two $, two @, an !, four numbers, seven letters, and it was cracked instantly. Because even though it could be really, really, long and really, really hard, when they look at it, it's actually a visual message. As that research keeps getting done to make predictive typing better and people fail to write English anymore in text messages and things like that. Those are all databases that cracking systems can use. It's all basically a word dictionary that could be downloaded from Google, from your phone, from wherever, and used as a simple dictionary attack as opposed to a very hard character by character numerical crack.

Bill: How big is that dictionary do you think?

James:  
[00:05:30]

[00:06:00] We did some checking. I can download the Websters New Word Dictionary with two megs and then the hacker dictionary that we were using was about 550 megs. 250 times the size of a Websters dictionary, and that was just English. Of course, a lot the crackers you want to have variants of English and foreign languages. There's also dictionaries that are being used now were you can get very simple downloads of movie titles, actor names, things like that. That's interesting because people think, that's not really a word it's a phrase, but it's loaded in a dictionary so that means even though its 20 characters long, it's still just a word from a dictionary point of view.

Bill: Or a title of a movie or something?

James:

[00:06:30]

[00:07:00] Title of a move, title of a book, all downloadable. You can basically pull a download of a title of every movie since I think it's like 1912 or something like that or some age old number. It's not very big, because it's just a text string. It's not like it's the IMDB database, although I'm given the understanding IMDB will also let you down load a text file of all the actor names. It's all stuff available to programmers, it's stuff available on the internet. Then, when you run it through a cracker, it's all stuff that's actually really, really useful. A very common password issue right now is winter, I think everybody loves Game of Thrones. Winter is definitely coming into all of the passwords that everybody has for sure. Really the issues that we found were habitual passwords, or people having a password policy, but the password policy is based on something visual, or even worse than that is keyboard patterns.

[00:07:30] Keyboard patterns have actually been a no-no for about a decade now, but people are still using them. It's kind of when you start from Z and you go up the keyboard, and then you start at 2 and then you come down the keyboard. People think that's got a lot of funky symbols and stuff in there. In reality, anything on a keyboard that's simple to type, that's also in the dictionary.

Bill: What was the visual one, that you were referring to? Are you talking about when you pick a map, point and click on here's point the bridges some of it?

James:

[00:08:00]

[00:08:30]

[00:09:00] No, visual being if you take the word message and you replace the, A with a @, or E with a 3. Visually in your mind it's still the same word. Unfortunately those are patterns that are really, really easy to get into a dictionary. When you have all of the developers working on that predictive typing and trying to help people write correctly, all of that can be reversed, downloaded into a very simple sheet of, "What are the most common things people do?" Now, all of a sudden you have a nice dictionary you can run as a password cracker. Visually that's what you're really talking about is, they're trying to type something that makes sense if they imagine it in their mind. Unfortunately, the computer can predict that better than the people can. It becomes pretty hard to make a good password and actually in this case a lot of the IT and other developer passwords were cracked quite fast. A lot of them were the ones that are supposed to be hard. A lot of them made no, weren't even words, it was just a list of symbols, but people who are going to type lists of symbols, will naturally have a keyboard pattern.

[00:09:30] You think about all the research keyboard manufacturers do to figure out how expensive a spring to I need for each one of these keys? Because, if I can save .20 by having lesser used keys have a weaker spring. People have done that research, so people know what keys are most used. From a cracking dictionary point of view that's not even difficult math. That's using data sets that are available, and a little bit of statistics, and then build that into the dictionary set. All those end up being pretty easy.

[00:10:00]

[00:10:30] Then, it comes down to, what are people suppose to do for passwords? When it was related to this incident, the interesting kind of hurdle to tackle was the issue that people were very reluctant to say, "I don't want something that's harder." The communication break was the fact that harder wasn't helping anyway. Because if we can crack really "hard" passwords that are 15 characters long in 20 seconds or less, making it harder's not really helping. Making it more convoluted wasn't really helping. What we found, the ones that were working the best were phrases intermixed with the numbers or symbols. What I mean by that, is whenever someone had a password where all the letters were together, and they were numbers or symbols before or after it, those are actually cracked really fast because the words were all together and it's just a question of tootling some symbols on either side of it to be able to crack the password. Some people would have really long passwords like steakdinner12. Okay that's just a word right there with a couple of numbers, not hard to hack at all. The ones that lasted really well were actually a lot short words.

Bill: Like sentences strung together?

James: Yeah, a sentence a phrase-

Bill: Yeah, sure okay.

[00:11:00]  
James:

[00:11:30]  
... a limerick what have you, but it was sentences where the number were in between the words. You would have to have word and then, be able to rotate characters with a number, and then another word, and then a number. That became very hard. Now, mathematically speaking any password can be cracked given enough time. Computers are getting faster obviously, but everyone is supposed to change their passwords. If you're changing your password the amount of time that people have to make use of a decent strength password is pretty finite. You don't have forever to sit there and crack something and play with it before it becomes unfeasible, and you can't do anything with it. A string of words with numbers in between become very useful. Especially if its more than one number. Mailman1 is not a very good password, right?

Bill: Yeah.

[00:12:00]  
James:  
You just try to rotate some numbers. They're gonna try zero and then you're gonna try one and now you have the password. It's not very hard. Intermixing those short words with a couple of numbers, another short word couple of numbers, short word couple of number. Not very hard to remember because you can have it be part of a sentence. But strength wise was much better than sitting there trying to replace four of the ten digits in your 10 digit password with the @ signs, and !, and $. It just wasn't really effective in comparison.

[00:12:30]  
Bill:  
Do the way these work, you might not know this, but does it have to guess the whole thing successfully or can it ... The way the computer works does it have to say, let's say Bob was the first part of it. Does it have to get the B the O and the B and then the computer says positive, positive, positive, like one, one, one? Or does it have to go the whole thing perfect?

[00:13:00]  
James:

[00:13:30]

[00:14:00]  
If people keep things up to date, it does not do that. For example in old operating systems in T4, Windows 2000, you could crack passwords in chunks. That's one of the main reasons that people who still have old Windows 2000, Windows 2003 servers and are basically dumbing down the security of the entire environment, because everything has to come down to the lowest common denominator. That's actually possible in some environments. It also depends on what system you're using, but if people are updating, patching, staying with relatively current. In some cases that's possible, but it's not as easy as it used to be. It is not a clear mathematical chunk of first you crack the first eight digits, then you crack the next four digits, then you crack the next five. It's not really chopped up that way, because as security's gotten better, how that's been chunked out has been scrambled. It makes it a little bit easier. Not all systems work that way, so it does somewhat depend on what system you have. We actually found that issue in this incident review as well, and it's because of legacy systems that are being forced to stay in the environment and quite literally it makes everything operate as if it's still that old. There are certain types of-

Bill: Because the authentication has to work off of that part, that member of the domain?

[00:14:30]  
James:

[00:15:00]  
Not because it bounces off of it, but because you have ... Think of it, if you have a pile of people who are biking together. That whole group if they're gonna stay together will go as fast as the slowest peddler. Same thing is true. That worst security server in your environment, everything has to be able to communicate all the way down to that level. It can't take advantage of the newer security changes. For example, there's something called a hash, very popular for passing the hash attacks. That's how some credentials are stored. If you have older systems you will have different types of hashes. Some of which are very, very easy to crack. It doesn't matter if your domain controller in Windows is updated if you've got a bunch of member systems that are very, very old. Everything has to come down and communicate to that level for everything to be able to operate correctly.

Bill:  
[00:15:30] That makes sense, interesting. Now, is there any way if a company can't force password changes and has to get to a standard that works, can they look at two factor as a means to mitigate something that might not be as strong? Would a weak, plus a strong second factor negate it, or do you need layers of both sides, really strong on both sides?

James:

[00:16:00]

[00:16:30] I does and it doesn't. A weak plus two factor gets rid of most of the external attacks that people fall victim to. As a good example, a number of the hacks were someone logging into Outlook Web Access, and then stealing an executives email, and then using confidential merger files, using wire transfer information, using W-2 information, and just ripping it out of email. Not really an interesting hack, they just logged in and forwarded the message to a fake Google address. Not really hard at all. Two factor helps with pieces like that, absolutely. That's happening more and more with fishing attacks. People fall victim to a fishing attack they give out their password in the fishing attack usually. Two factor authentication prevents someone from then using that password against them.

Bill: Against them, yeah.

James:

[00:17:00]

[00:17:30] However, once things are inside the walls, malware's ransomware is key there. Password strength is still actually really important, because if I've got a mini cracker and you download my malware, or Trojan, or ransomware. I can sit there and try to crack some hashes and if the passwords are all weak I'm going to get them probably faster than your behavioral analytics will notice you have a weird process running. Now all of a sudden I'm an administrator. I get to escalate my privileges and run around your network, because of that point of view that says, "Hey, two factor authentication means my password can be weaker." I think that's another piece that really needs to be explained is, IT is traditionally really bad at telling people to make their password complicated. They really need to be telling people to make their password interesting. And interesting password you can remember. It's going to be longer and it sounds less scary to tell someone, "Make your password interesting." Versus, "Make it more complicated."

Bill: Right.

James: Complicated's a negative word.

Bill: That could almost be like the title of this blog post. Make your password interesting, don't make it hard.

James: Make your password interesting. I had one password that remained uncracked in this one case. It was a password created out of annoyance that was basically, "I'm not going to do it". It's not really hard to remember the statement, "I'm not going to do it."

[00:18:00]  
Bill:  
No symbols or anything?

James:

[00:18:30]

[00:19:00] It was really, really long it had really weird combinations of words. You would have to run a very complicated algorithm on top of dictionaries to get all of those words. Sadly, that was really effective. Now of course, you have to change it, but it's not really hard to come up with a sarcastic sentence once every 90 days to change your password and have it be pretty decent. Now, realistically we still think there should be numbers and symbols in there, but just as an example sometimes a raw combination of words. People don't do that. Intrinsically people are lazy, that's why you have passwords we find like, welcome1, because that was what they were given when they started, and ever since then they change it. For example, welcome9. That guys worked there for 9 years. Every year he changes his password once. It's funny but that's really what you find.

[00:19:30] We found a lot of patterns. It was kind of interesting as we went through having hacked the passwords, we could then sit there and say, "Well, what do the passwords tell us about the environment?" You would be really, really surprised where, as an example, we found 51 accounts that had almost the same password because people who are similar, think similar. If they've had similar training and had similar experiences, the chance of them using basically the same password is really high. We found about 50 with literally the same password.

Bill: Oh, my gosh.

James:

[00:20:00]

[00:20:30] Either IT has a bad habit of setting all new users to only on password and then they really don't vary much from there, or you have groupthink. Groupthink is a standard business issue. You put five developers in a room, you ask them a question you'll get a developer answer. That's groupthink. We've found tons of issues like that. For an example, we found 51 that were all one word with 1, 2, 3, 4, after that one word. That was 51 different peoples passwords. We found another common word that had about 27. We found that another 60 that were either summer, or winter, with a four digit year afterwards. Depending on the physical location the number of, it was very interesting there were for Maryland a lot more Ravens fans than Redskins fans. That's in there. In reality, it's just a word. Ravens is just a word, Redskins is just a word. You can kind of look at the data and then start seeing how people think, but realistically when we find the same issue with in house developers, outsource developers, IT folks themselves. You start to realize, "Okay, what people think is good, is incorrect."

[00:21:00]

[00:21:30] We definitely found that, because if you got a 12 digit password and six or seven digits are symbols or numbers that supposed to be really hard. When it's instant, that's not really, really hard anymore. Unfortunately computers at this point are learning way faster than people are. Keyboard patterns and natural character replacement and things like that, it's just too easy. Some of the malware's now, it's really interesting they will run the cracker on the processor in your graphics card. It won't even slow down your PC because most people don't even the CPU in their graphics card, because they're not doing 3-D video editing while playing with Microsoft Word.

Bill: I think the GPU is faster anyway.

James: The GPU is faster anyway, so it'll just eat that. There's nothing that's gonna go tell network operations, "Hey, the CPU on that PC is spiking." Because they're not even using the CPU. Like you said, it's faster.

Bill: That's pretty clever.

[00:22:00]  
James:  
That really becomes an issue, because once you start cracking passwords, then you can pretty much do whatever you want.

Bill: Now, we've had recently had someone, this happened to a couple of different accounts. The wire transfer, I guess you could call it wire transfer fraud.

James:  
[00:22:30] We've seen wire transfer fraud, we've seen the W-2 scam, and we've seen basically a lame request for PayPal Payment. We've seen all three of those, quite often.

Bill: They're fundamentally based on I mean, the hacker has to get access to that password, and then has to lie as a legitimate user.

James:

[00:23:00]

[00:23:30] Exactly. Originally these were considered spear fishing attacks and the better spam filters got a blocking them, and the more users were trained not to fall for fishing attacks, the less those were able to work. Usually the W-2 scam would start as just kind of a blind external fishing attack. Really, it was trying to catch people with the idea of, here's a fake looking email that came from the CEO. The CEO's yelling you've been trained to respond when the CEO yells therefor you do immediately what they say and you don't really think about, does this email look weird? Is the address mistyped? Little things like that, next thing you know somebody sends all the W-2's from payroll out to China and then you have the issue of data breach. Really that has to do with tax fraud, which is really a good money make, very popular. Simple IRS tax fraud at that point.

[00:24:00] Now, what they've found is if they want to make it even, basically if they want to get even the best trained people to fall victim to it, the best way to do it is actually send the email from the person. Once someone loses their password you just log in as the user and what they're doing is they're looking at the sent mail, they're finding the conversation in flow. Instead of sending a random email, they're just replying to an existing conversation. What's safer than continuing a conversation you've already started. Their replying to that existing conversation, and then they inject the, "Oh, sorry we changed our bank number you didn't know that. Quick here's the new banking number." It's in the flow of a conversation about an overdue invoice, or a payment you have-

[00:24:30]  
Bill:  
Sure.

James: ... or a deposit you owe. They're staying in ... What's a good word for it? They really staying in character.

Bill: Contextually, within the message, so it doesn't seem as it's like a request from left field?

James:

[00:25:00]

[00:25:30]

[00:26:00] Exactly. In this case for example, what they can do because they're in and they're doing all this with public web access, so it's not like they high jacked a PC. They're just using a web browser. They're going in through Outlook Web Access, or OWA, or Outlook.com, whatever you want to call it. They log in as the user and when they got to reply, they just change some of the return addresses. Then what happens there is when they send the response of, "Oh are you sure you want me to change this routing number?" That's going to now, a fake account who's going to reply, "Yes." When you think of spam filters, spam filters are usually set up that if you sent that person an email they must be legitimate otherwise why on earth would you email them? In this situation they get the victim to send the first email, which helps them bypass spam filters. Then, lifts away the issue from the persons whose password they stole. Because now they're communicating directly with the victim and the person whose password was lost or stolen they're really the patsy for the criminal. They are not the person who's continuing to be used, because they don't want to lose the ability to send emails as that person.

Bill: They're sending outbound. They're picking up on the Outlook Web Access, which for the nontechnical folks, which is probably not a ton, for the nontechnical folks they don't have to be inside the network for that.

James: Correct.

Bill:  
[00:26:30] Which is a huge bonus for a hacker. Then that person is reply all, and so it goes out to the potential victim to ask them to pay their bill with a new routing number and potentially a new .co, or .whatever, domain name, which is just subtly different. Then there's no need to more back and forth communication right?

James: Correct.

Bill: Because now they're directly on a lending page to pay the bill.

James:  
[00:27:00]

[00:27:30] Typically, what they're doing there is, in terms of the persons account they're spying on, they're specifically not touching anything. They're just looking at it, so it doesn't trip any flags. They're very, very, very carefully saying, "I don't want to put the credentials I've stolen at risk, so I'm just going to look and then when I take action, I'm going to do it out of band." For example in this one case even though they stole person A's credentials when they communicated. They communicated as person B in the chain. That way, anybody going to look into it would go look at person B. See that everything's fine with person B and say "Okay, you must be getting fished from the outside world." They never really refer back to person A, even though that's where the conversation started and how they made everybody feel safe.

[00:28:00] Because if they're criminals, right? Standard criminal activity, if you've got a mark that you can use, you don't was to destroy or spoil that mark. You want to protect them, so you can keep using them over and over again. That's the nature of these kind of more advanced fishing attacks is that ability that says, "Oh, if your silly enough to lose your password, you don't have two factor authentication, your password was decidable, you fell victim to a fishing attack and gave someone your password." If those things are true then they don't want to spoil the fact that they can keep using you over and over again, kind of like blackmail.

Bill:  
[00:28:30] The net-net of this particular example is from a password point of view, if we had infinitely take the percentages way down by following the interesting not hard model of password creation. Then, because of the external access for owa having obvious second factor in, would actually make it very difficult as well, by having the second factor right?

James: Absolutely. Two factor authentication solves 90% of the world’s problems for remote access.

Bill:  
[00:29:00] Are there any limits per-say with Office 365 at this point. They have to be, is there any limits from this type of attack being whether is on prem, off prem, or a straddled hybrid email environment?

James: Not really. You can protect it any direction and no direction is safer than any other unless you purposefully make it be protected. None of them get protected by default.

Bill:  
[00:29:30] The owa doesn't necessarily need, owa is client dependent so you don't have to be on Windows 10 to have two factor with Office 365 or any other special browsing capabilities.

James: Correct.

Bill: It's just-

James: Nothing special.

Bill: Actually the dumber that browser it's Microsoft's best interest to have it as dumb as possible, right?

James: Correct.

Bill: So they don't need to have a client side there. Okay, that's interesting.

[00:30:00] Well, I think that's good. That's good for today. It think that's a ... Haven't done a pure tech oriented, super practical, passwords, two factor, good case study, good practical example of how we've observed this happening. I think this is gonna be very useful for everybody listening. Until next time everybody have a great day.