

It Happened One Friday - Answers and Winners

At long last, we've completed final judging on the It Happened One Friday challenge. I apologize for the delay, but things have been very hectic here. We received a huge number of really top-notch entries in this challenge, and reading through every one of them and whittling them down to our final winners was fun but incredibly time-consuming. However, I'm really happy with the final results – the technical and creative winners did some awesome work, as did many others worthy of an honorable mention.

Given the unusual nature of this challenge, before I announce the winners, I'd like to provide a little context to describe why Matt Carpenter and I wrote it the way we did. If you will kindly indulge my explanation for just a bit, I'll describe for you a little bit about the process of writing these challenges, and how this one in particular came to be. Alternatively, if you are impatient, you can skip these author's notes and jump to the announcement of the winners by clicking [here](#).

--Ed Skoudis, InGuardians

Author, Counter Hack Reloaded

[del.icio.us Slashdot It!](#)

[Discuss in Forums {mos_smf_discuss:March 2008 - It Happened One Friday}](#)

ChicagoCon 2008f

Winners Announcement and Author's Notes for It Happened One Friday

By Ed Skoudis

Whenever I write one of these challenges, I try to make each one look and sound different from all of my earlier challenges, present its narrative in a different way, and analyze a different kind of hack that integrates with the movie theme in a unique fashion. I don't want the challenges to be formulaic, always following a simple pattern of a paragraph or two about a hack, the same type of overlaid movie theme, a joke here and there that plays on the movie idea, a presentation of the evidence, and then the same set of four stale questions at the end. I like each one of the challenges to be as different as possible from all of the others. However, even with their differences, all of the challenges have one thing in common – they tie together technical ideas that are useful for security professionals along with a movie or TV theme. And, It Happened One Friday was no exception.

So, what was the movie theme for this one? Why, it was Mel Gibson's The Passion of the Christ. Some of you are likely thinking, “What? You've gotta be kidding me.” Nope… that was our theme.

The idea for writing it came to me back when the movie was originally released four years ago. A couple of days after I saw the movie, I hit upon the idea of making it into a hacking challenge. I thought about having a computer system represent Jesus, totally shredded by an attacker. It seemed somehow blasphemous for Jesus to be a Windows box, and more appropriate for him to be a completely pristine Linux system. As I figured it, the bad guy (Satan, of course) would destroy the machine causing it to crash hard on Good Friday. Then, mysteriously, it would reboot completely renewed three days later. I also wanted the challenge to be entirely respectful from a religious perspective, a humble devotional, a meditation of Christ's suffering on the cross, told from a hacker's perspective through the eyes of technology. Oh, and I wanted it to be as theologically "correct" as I could make it, based on my own layman's understanding of the transaction that occurred on Calvary a couple thousand years ago. I even thought of writing it in King James English. I also wanted it to have some deep technical command-line meat, from a Linux perspective, testing readers in deciphering how thoroughly an attacker could destroy a machine in the ultimate denial of service attack. Those were my rather preposterous goals for the challenge. In keeping with the challenge-naming scheme I've used in the past, which involves substituting the word "hack" inside a movie name, I even gave the challenge a working title: The Hacking of the Christ.

But, that title seemed too disrespectful and in your face. Further, the original movie itself was incredibly controversial for all kinds of reasons I don't want to get into here. Quite honestly, I didn't want to jump into that fray by using it as a challenge theme. I also didn't feel up to the task of writing the prose necessary to strike the balance between appropriately representing the story and giving a good technical challenge for people to contemplate. After some deliberation, I shelved the whole idea, thinking that I couldn't pull it off properly. I've got ideas for dozens of challenges that never really make it to final form*, all aborted somewhere along the route to completion.

Flash forward almost four years. Last Christmas, I was working on the Frosty the Snow Crash challenge. I was in my Frosty-writing groove, thinking about the upcoming Christmas holiday. I had been spending a lot of time in the run up to Christmas contemplating the Incarnation. The Son of God, wrapped in flesh, dwelling among us as a completely dependent babe, lying in a filthy feeding trough, come to change the very nature of all humanity's relationship with God. That's what Christians believe. Talk about preposterous. Why, it's enough to make you melt like Frosty the Snowman. I started to think about how Frosty, in his own story, had some interesting Christ-like qualities. He sacrificed himself to save Karen the little girl from a frozen death, but was risen shortly afterward by Christmas magic. Frosty as Messiah figure. Yeah, I'm weird; OK?

Later that month, I went for one of my morning walks. During my walks, I think about stuff — technical concepts, ways to express ideas, relationships, Life, the Universe, and Everything**. Sometimes, I listen to podcasts during my walk. One of the podcasts I listen to on occasion is called Watching the Directors, which analyzes movies from a critical perspective, dissecting a film director's career to see how themes interweave through his body of work. I'm an amateur moviemaker myself, stitching together simple movies of my kids for fun. I'd been studying how movies can manipulate audience emotions, and found this episode of WtD particularly interesting, with its focus on Mel Gibson's work, including Braveheart and Apocalypto. Inevitably, during the podcast, Gibson's The Passion of the Christ came up.

I listened very carefully. Neither Joe nor Melissa from WtD liked The Passion of the Christ as a movie. To explain why, they centered on the idea that The Passion doesn't really work as a movie, which is a story-telling medium based on character development. The Passion doesn't really tell a story, and there isn't any character development. It's a completely minimalist work in that sense. Instead, it's really a contemplation on the Crucifixion. It's more like a painting than a movie: a work of art that grabs your attention and moves your emotions, but not a story. If that's the case, then why did so many people feel so moved by it? Joe and Melissa explained that it was because the movie simply amplified the story that each person brought to it. Many devoted Christians loved the movie, because they looked at it and said, "That's what He did for me." Other people found the over-the-top violence and blood appealing, resulting in what they viewed as a fascinatingly artistic snuff film. Filmmaker Quentin Tarantino ranks The Passion among his favorite films because of its intense visuals and the nature of its gore. Part of the reason for the movie's success was that it didn't have the trappings of a normal movie — no real plot, no character development, no prose; It merely amplified what you brought to it.

After listening to the podcast, I continued my morning walk, with Frosty the Snowman, the unfathomable mystery of the Incarnation, and a better understanding of what The Passion of the Christ movie really was, all swirling around in my

head.

And, then, all at once, it hit me. I now knew exactly how to do the Passion-themed challenge. No prose. Just some system log files that display extreme violence against a Linux box. No story, other than the story you bring to it... possibly the greatest story ever told. But I wouldn't be telling it, the log files would. No overt religious references, just subtle concepts woven throughout. If you were the type of person who wanted to read it in a purely technical fashion, well, we'd have some deep command-line kung fu to tickle your fancy. If you were a devoted Christian, perhaps you'd pick up on the theme and be moved by it. If you were a super creative person not into our theme, you could superimpose your own tale of destruction and woe on our log files. Oh, and, we'd release it on Good Friday, part of the big hint underlying it all. But, above all, we wouldn't force you into a movie or religious interpretation. You could bring your own story to it, just like The Passion of the Christ. In fact, it occurs to me now that the actual Gospel story itself is minimalist, building on and amplifying what each person brings to it from his or her own walk. Oh, and the questions at the end of the challenge would be totally minimalist as well – What happened and why? The “Why” left things open for interpretation, challenging people to try to figure out the motivation of the various players in our challenge. I abandoned the original title too, as it was too overt. Instead, the name would simply be It Happened One Friday. That basic framework for the challenge and those concepts all came to me in about 30 seconds.

Then, continuing my walk, I started to brainstorm the commands that we'd see in the shell history file. Clearly, Satan would have to chmod 666 something, just to work in the number of the beast. I started to think about how someone could destroy a Linux machine progressively, starting by just scratching at it, but eventually completely hosing it. In fact, I wanted the bad guy to even trash the BIOS from the command line before blowing the whole thing up. I wanted severe damage, to make the mystery of the pristine reboot even bigger. To achieve such devastation, I needed someone with extreme Linux skills as a collaborator on this one, someone who would understand the goals and theme of the challenge implicitly. Instantly, Matthew Carpenter came to mind. The guy is brilliant, from both a technical and narrative perspective. He'd be perfect as a co-author. I was delighted when he agreed to work on it with me.

A couple of weeks later, I called Don Donzal, telling him that Matt and I were going to write him a new challenge for the Ethical Hacker Network. Don liked the idea and even created some really cool artwork for the challenge, a computer that disappeared for 3 seconds (a symbol of the 3 days between death and resurrection), with a subtle cross on the front of the machine.

Unpublished Artwork That Gave Away Too Much

Unlike our other challenges, this one had a firm deadline for posting – Good Friday 2008. About three weeks before the challenge was to be posted, Matt and I started brainstorming about it, spending our morning walks on the phone with each other putting together the pieces. I hit upon the idea of deleting bash, because Satan surely wouldn't want the “bourne again” shell sitting around. Matt decided to do the root pivot, which was very cool, allowing the bad guy to take pot shots at the file system while keeping the box running. I wanted to show the imputation of sin to the victim, so I had Satan mark a bunch of blocks as bad and made the file system dirty. Matt fine tuned the log file timestamps so that the machine went down at the appropriate time on Friday and rebooted at sunrise Sunday morning. We even researched various religious scholars's views about the actual dates and times for the events, applying the appropriate offset for UTC time based on events transpiring in Jerusalem. We also incorporated INRI on the defaced web page of the victim Linux machine, just like the sign hanging above Jesus's head on the cross. The message on the website was a base-64 encoded form of the Hebrew language used on that sign, something you could look up via Google and Wikipedia. The majority of the responses we received were able to figure out this message, but were puzzled as to its relevance.

The log server was called johnboy, a reference to the Gospel of John, dutifully recording the action. The bad guy was called “luz,” a shortened and modified form of Lucifer. He used a Netcat listener on TCP port 54742, Matt's numerical interpretation of “SATAN”, with 5 being an S, 4 being an A, and so on. Yeah, that's a bit oblique, but what are you gonna do? The Netcat listener allowed the attacker to enter e-mail addresses into the “mail” program remotely so that he could announce his victory over the target to the world.

We also wanted the bad guy to make mistakes, both to indicate that he wasn't a script, and that he had some

personality, admittedly a diabolical one. He mistyped commands, such as `uname -a`. Sometimes, he forgot that he had already destroyed pieces of the file system that he tried to use later, like trying to create a file in `/tmp` after he destroyed the directory structure where `/tmp` formerly resided. He even tried to look at the man pages after trashing them, making him cuss a little bit at the system for his own mistake with a hearty `fsck you`. He quotes Trinity (yes, another religious thematic hint) from The Matrix movie by logging the message `Dodge This`; just before he zeros out the memory of the target machine, making it die.

As a final message wrapped in the challenge, we chose the IP addresses of the machines very carefully. Looking at the logs, the target machine itself was 242.229.249.233. We know from the defaced web page that the Hebrew language was involved, so we could look up the Hebrew ASCII at <http://www.ascii.ca/iso8859.8.htm>. The results of the decimal-to-Hebrew conversion is: `ÙéÕâ`. What's that? Well, remember that Hebrew is read from right to left. Thus, reversing and translating, the target was called Yeshua, a more Hebrew-oriented name for the Man on the cross. The attacker was 228.229.228.233, also known as `ÓéØß`. Again, reversing and translating, we've got `Ha-Satan`, or `The Satan`. Finally, the initial login to the machine happens from 228.229.228.233. That's the Name of the owner of it all, and is the clue for how the whole hack starts out.

At the beginning of the log file, we can see the owner of the system logging in via ssh and altering `inetd.conf` to allow inbound telnet. He then telnetted into the box, `su`ed to root, typing in the root password, and exited. The owner did this on purpose so that the bad guy could sniff the cleartext telnet session, gathering a userID and password, as well as the root password. The owner introduced the vulnerability, indeed the only vulnerability, in the target, allowing the bad guy his chance to cause destruction. It was all part of the owner's Big Plan. Some of the submitted answers did pick up on the idea that the owner was complicit in the attack. The attacker was shocked at his good fortune, and didn't believe it at first. That's why he scratches around to make sure he really has root early on in the attack. And the rest, as they say, is history.

So, that's how this challenge all came about, and some commentary about the stylistic elements. I wanted to peel back the curtain to give you a glimpse of the process behind creating this one. For a really good technical description of each item in the shell history and log file, I heartily recommend that you read both the technical and creative winning answers, which we've linked to below.

Finally, thanks to Matt for all his hard work, thanks to Don for hosting it, and thanks to everyone who worked on answers to our questions. Oh, and thanks to 242.229.249.233 for making it all possible.

--Ed Skoudis.

* For example, `Dude, Where's My Hack?` will likely never see the light of day, nor will `Debbie Hacks Dallas`;

** Yes, I know about Douglas Adams' noted atheism.

Technical Winner

Alexandre Dery: This entry is an awesome piece of analysis. The Force is strong with Alexandre! Great work.

Honorable Mentions:

Rick Hayes: Rick provided very good, detailed, command-by-command interpretations.

Frode Mangseth: Frode really analyzed the mistakes made by the bad guy quite well -- typos, trying to use commands he'd already deleted, etc. Nice analysis.

Kaito: Very good work. He nailed many but not all of the commands. Kaito explained many of the technical high points of the direction of the attack and reasons for big components of it.

Kern: Very solid understanding of some of the commands.

Dmitry: Likewise, a good description of many of the commands.

Nicolas Krassas: Like a few others, Nicolas pointed out that the ! in the e-mail subject line would break the command. Good point! Also, Nicolas noted that, "The attacker is most likely somebody that has a relation to the victim because his persistence on erasing the machine is really strong." I’ll say!

Radislov: This gent got the chmod 666 reference, saying, "[the] malicious user is probably a death metal listener, which is in contrast of his previous religion sign." That’s an interesting juxtaposition, isn’t it? A mixture of the sacred and the profane. Radislov also discovered the problem with the ! in the mail command. He also noted that, "It seems like owner user is real owner of the system. It could be, that owner set up an no logging telnet and added user luz with UID:0." He then posited a password guessing attack... that’s a very good interpretation of the facts at hand! Finally, Radislov comments, “That Yoshua was really a g33k." I'd like to think so.

Kees Leune: This solid answer noted that attacker checked the web page defacement using wget to localhost, which wouldn't be filtered and is thus not a very good way of verifying the defacement from the outside world. Excellent point.

Israel: This gentlemen’s answers were very well thought out. Excellent work. At the end of his write-up, commenting on the reboot of the trashed system, he observes, “The server is back? Not sure what happened here, Got entries for ok startups… In the end this machine is pretty much gone.” Yes, indeed. It was pretty mysterious.

Brian Wilson (Not the EH-Net Columnist): Brian provided some excellent work, as usual for this really smart guy. In particular, he identified some minor miracles in the write-up, such as telnetd having the same PID for two different connections 30 minutes apart. What can I say? It was a miracle... a cut-and-paste miracle. In fact, as Brian says, "The log entries were duped.” He also notes that the remote ssh port was the same for two different connections as well. Again duplicated logs, he posits. Good catch, Brian! You’ve identified a glitch in the Matrix that Matt and I constructed. It’s not like we’re writing scripture, you know. In fact, we were testing you, and you passed with flying colors. Yeah, that’s it.

Interestingly, no one ever called us on how the shell history was exfiltrated from the destroyed machine so readers could view it as evidence. I thought someone might bring that up. I’ve noted in writing these challenges that we don’t always have to say how evidence comes into being. Sometimes, we can just present the evidence and people work on it. As a writer, I have been able to rely on the omniscience of the narrative, and readers seem to accept it. I used this same tactic in the Frosty the Snow Hack challenge, with the magician’s typed commands just appearing in the narrative without any description of how the reader got access to those commands. In my early challenges from years ago, I used to struggle for contrivances to explain to readers how the evidence they were viewing came about. Since then, I’ve learned we don’t always have to explain the origin of evidence in these challenges.

Creative Winner

Walid Shaari: This submission was a really good technical answer with some wonderful creative insights. For example, Walid realized the reference to 666 as the number of the beast, and correctly translated the Hebrew from the defacement.

Honorable Mentions

Ryan Linn: This very clever answer posited that an Israeli hacker broke into a Palestinian server to do some damage. It was quite interesting!

J Bruce: This very curt answer was brilliant in its minimalism. Here is his entry in its entirety: "The IT Admin came over to the employees desk and said, "The customer would like 5000 lines of code to print Hello world to the screen. Make them happy." The employee's head turns backwards and he begins to bang his head on the keyboard. The result is the combo of char's you see on the screen. 1000 seconds later the employee gets another Monster drink and plays pong."

Dan Roberts: Dan's answer was really one of the best ones of all. He did very well technically and identified the entire theme and narrative perfectly. He did great work... but he submitted his entry long after the deadline, so is not eligible to win. His follow-up thoughts are gripping, and Dan ends his write-up with the poignant thought, "In the end, perhaps the explanation of this story (like the one it parallels) just needs a giant leap of faith to accept. :-)" Amen to that.

And, the Random Draw Winner:

Kees Leune. Congrats, Kees!