

Charlotte's Web Site - Answers and Winners

del.icio.us

Discuss in Forums {mos_smf_discuss:Feb 07 - Charlottes Web Site}

Seeing as though Ed Skoudis is not only the host of the Skillz H@ck1ng Challenge, but he also wrote this one. So, I figured I would break from the norm and introduce this one myself. Why you ask? Doesn't Ed do a good enough job? Of course he does. What Ed will not do is toot his own horn. That's where I step in. The effort Ed puts into these challenges is indicative of who Ed is as a person. He does not get paid by EH-Net. All he gets are links back to his company, a mention of his work with SANS and maybe a few more of his books get sold. Yet when he writes a challenge, it is thought provoking, has depth beyond just the narrative and above all shows off his sense of humor. When the challenge is complete, he goes even further by offering insights into his thought process (which should interest all EH-Net readers) and windows into his creativity which is evidenced by the details contained in this article. So Ed... here's to you, my friend. TOOT TOOT!! Or should we say OINK OINK!!

And now onto the answer and winners. Drum roll please...

Salutations! It's time to announce answers and winners of our Charlotte's Web Site challenge. Let me start by saying that you guys submitted Some Answers. They were Terrific! Radiant! Some of them were even (dare I say it?) Humble. And, for the fans of the Charlotte's Web book, a lot were Pre-shrunk and Crunchy.

--Ed Skoudis, Intelguardians

SANS Instructor

Author of Counter Hack Reloaded

Skillz Sponsored by Core Security Technologies

Before announcing the winners, though, let me provide you with my view of the answers.

1) What is the significance of various numbers in the story, including the speech patterns of the goose and Templeton?

If you couldn't tell, I rather like number play. But, I did not want to require you guys to do any heavy mathematics in a challenge, so I embedded various numbers in the story that are interesting and fun. Let's look at each number in the story.

The Wilber auction ID was 3141592. Those are the first 7 digits of pi, that magical number that you get when you divide the circumference of a circle by its diameter. Note that the auction ID is not equal to pi, as a few of your answers mentioned. It is a close approximation of pi. The beauty of pi? Its digits do not seem to repeat... ever. It's a transcendental number. Think about that... what a miracle. The universe has put before any sufficiently sentient beings a nifty challenge: if you can measure the properties of a circle accurately enough (or figure out an equation to calculate them even more accurately than you can measure them), you'll get this magical number that shows up everywhere, including, to a rough approximation, in the auction ID of a pig on Meat-Bay. Oh, and also, please note that I put in 7 digits of pi, while for all the other numbers, I used 6 digits. Why? The number 7 itself is used in numerology to indicate completeness, with some thinking it to be a number closely aligned with God. And don't get me started on the number 6.

The next number in the challenge was the first bid on Wilber, which was \$ 1,618.03. That happens to be the first 6 digits of the Golden Ratio. For centuries, artists have believed that this mysterious ratio (approximately 1.61803) makes things seem well-proportioned. To the human mind, this ratio just plain looks good; it's aesthetically appealing and appears throughout history in paintings, sculpture, architecture, and even music. Heck, even the post cards you receive in the mail are often formulated so that one side is 1.61803 times larger than the other side. Some believe that this ratio looks good to people because it stands at the boundary point between symmetry and asymmetry. Others think it might have to do with the proportions of the human face, with some medieval folks believing that the perfect human face would have these proportions. Sometimes called Phi, this number is related to... Well, let's not get ahead of ourselves. We'll look at the relationship of the Golden ratio to another number a little later.

The next number of importance in the challenge is the follow-up bid on Wilbur: \$ 2,718.28. Those are the first six digits of the mathematical constant e, also known as Euler's number. That's the base of the natural logarithm (ln). Like pi, e is also transcendental, and, gosh, is it important. It has the interesting property that, when exponentiated, it creates a graph whose derivative is the same as the graph itself. The slope of the curve at a given point is the same as the value of the curve at that same point, for the entire graph, or, in other words (symbols, actually):

Whoa... Amazing. And, here's an equation that really baked my noodle when I first learned it back in school:

Folks, do you realize how incredible this really is? This simple equation has deeply spiritual implications for me. It makes me feel like we do live in the Matrix. It not only includes e and π , but also i , the square root of negative one. I mean, it takes a pretty bizarre mind to even conceive of making legitimate (but imaginary) numbers out of the square root of negative numbers. Beyond e , π , and i , this equation also includes the multiplicative identity (1), but used in an additive sense (an identity is something that leaves the original entity the same after an operation is applied; for multiplication, the identity is 1, because x times 1 is x). This equation also includes the additive identity (0). All of this is bundled up in a nice package. So, think about this; Humans have known about 1 since the very beginning, and the number zero began to be used in ancient times. Then, different mathematicians working on very different problems over the centuries independently discovered π , e , and i . Then, pushing the envelope, humans discovered this beautiful relationship between these powerful numbers. Logic and the universe itself have conspired to make this so. You ask, "Is there a God?" Well, I respond that e to the πi plus 1 equals zero. It gives me chills, I tell you.

Now skeptics might say that this equation is merely a more complex way of saying something like $1+1+1=3$. That, says the skeptic, is not a sign of God. I respond; are you so sure?

But, as fun as such musings are, we've got other fish to fry, or bacon to burn, as the case may be; Moving on to our next numbers.

We now look at the speech patterns of the Goose. She repeated syllables again and again, but notice how they change with each thing she says. She starts by saying the syllables "urvy" twice. Then, she says "eemly" three times. The pattern in her stuttering is 2-3-5-7-11. Those are the first five prime numbers. Each of them has only two factors: 1 (there's that multiplicative identity again) and themselves. Note that 1 is generally not considered to be a prime number by most modern mathematicians. If 1 were considered prime, many proofs would have to say, "For all prime numbers except for 1"; By ruling 1 out, we don't have to carry that linguistic baggage in our proofs. Because 1 is not considered prime, the Goose doesn't use it (also, it's kind of hard to represent a stutter that uses only 1 syllable; It's not a stutter at all, you see?) Instead of counting the entire syllables which were repeated, some of you counted only the repeats themselves, coming up with a series of 1, 2, 4, 6, and 10. Each member of this list is one less than the first five primes, of course.

And that brings us to our last numerically significant item, the speech pattern of Templeton the rat, who says, "O, I do not enjoy starving progressively!" The O was a clue here. I try to avoid typos in these things (believe it or not*), and the interjection "Oh" usually has an h in it. I purposely omitted it as a hint that something was going on with Templeton's speech. Count the number of characters in each word, and you'll see that it has the sequence 1, 1, 2, 3, 5, 8, 13. This interesting sequence is created by starting with 1 and 1 (that darned multiplicative identity shows up everywhere, doesn't it?). Then, for each subsequent number in the list, we add the previous two numbers together, yielding two (1+1). The next number is 1+2, which is 3. Then, 2 plus 3 is 5, and so on. This interesting progression is known as the Fibonacci series, and is very magical. (By the way, the Fibonacci series usually starts with zero, but making a word with zero letters is kinda hard. If it makes you feel better, you can imagine that Templeton actually started his sentence with a zero-letter word before the "O", but when printed out, a zero-letter word is invisible.) Either way, this series shows up all over the place in nature, in flowers, pinecones, and even the sticky globs on spider webs. Charlotte herself would surely approve. And, quite interestingly, this series ties into the second number we described above, the Golden Ratio. By carefully manipulating the Fibonacci series, one can derive the Golden Ratio, as defined in detail here.

So, in the end, of the numbers in the story, e and π are tied together. So are the Fibonacci series and the Golden Ratio. The outlier here is prime numbers. Nature is full of numbers, especially Fibonacci numbers. But, primes are quite rare in nature, making them even weirder and more special.

2) How had Charlotte and the Geography Ants fooled Lurvy's integrity-checking script?

First off, some of you noticed the significance of the Geography Ants. For the others, did it seem weird to you that these strange creatures just show up in the challenge, coming from neither the movie nor the book? As it turns out, their name is actually a somewhat famous anagram associated with a challenge posted some years ago on the Internet by the mysterious JonnyX. The challenge included a photo of the word "Geography Ants" hand written in pen on the belly of a woman. (Some elements of that old challenge are described by crypto-queen Elonka here and here. And, parts of the challenge itself are located here, but they have some bad words in them, so beware.) The phrase

“Geography Ants” is an anagram of the word “Steganography”, and was a hint in JonnyX’s challenge that Stego techniques were used, so I decided to use the same hint in my own challenge. Note, though, that I did not resort to printing my hints on a woman’s belly. That shows huge restraint on my part, doncha think?

Secondly, speaking of references, many of you noted the tie-in to the TV show 24. In our challenge, they had to fake Charlotte’s death because she was implicated in the killing of a diplomat at the Chinese Consul. The same thing happened to Jack Bauer (no relation) in Season 4 of 24. In the TV show, only Tony Almeida, Michelle Gessler, Chloe O’Brien, and President David Palmer knew the truth. And, for that truth, all paid for it with their lives (except Chloe, who almost did). Somehow, though, President Palmer knew not only the truth about Jack Bauer, but also the truth about Charlotte the Spider.

Anyway, Charlotte fooled the integrity-checking script by using an MD5 hash-collision. She relied on a tool created by Dan Kaminsky, known as confoo. This perl script retrieves two different web pages (call them A and B), and creates two new pages (call them A’ and B’) that look almost identical to the first two pages, but which have a hash collision. Confoo does this magic by starting with two blobs of bits (call them X and Y) that happen to have a hash collision. Then, we take X and append to it a new web page that contains some javascript and the entire contents of both A and B. The resulting page we’ll call A’. We do the same thing with Y, appending the same package of javascript, A and B, and we’ll call the result B’. Because we are appending the same package to two values (X and Y) that have a hash collision and are increments of 64-bytes in length, the resulting two pages (A’ and B’) also have a hash collision. The hash algorithm becomes synched inside of X and Y, and sticks for the package appended to both. In other words, the collision is preserved by the appending action. When loaded into a browser, this javascript logic included in both A’ and B’ decides whether it should display the first page (A) or the second page (B) that are embedded in it on the screen. This decision is based on whether X or Y is included in the front of the package. So, A’ and B’ have the same MD5 hash, but one looks like A when rendered in a javascript-enabled browser, while the other looks like B when rendered in the same browser. But, both A’ and B’ have the complete content of both original A and B pages. They also have the same javascript. The only difference is that one has X up front, the other Y.

Of course, as several people pointed out, if you view the source of the pages, or view them in a browser with javascript disabled, they look very different. Also, the source is very ugly, an artifact of the encoding done by confoo. But, Lurvy obviously didn’t look at the source, as he didn’t notice the discrepancy.

So, Charlotte created a hash collision of two pages, A and B. The first (called A’) looks like Lurvy’s original page, with an ad for Wilbur. The second (called B’) is an anti-advertisement for Wilbur.

3) Why did Charlotte have to change the website before the integrity-checking script ran for the first time? Why couldn't she deface it later?

You can use current hash collision technology to forge two brand new entities A’ and B’ at the same time that have an MD5 collision. But, with the current technology, you cannot create a brand new entity that has a hash collision with something that previously existed. In other words, you cannot create an A’ that has a collision with a previously existing A. Not yet… perhaps someday you will be able to do so, and that will be the death of MD5.

So, Charlotte had to change the page before the Lurvy’s script first imprinted on Lurvy’s original page. She had to sneak in and alter the page, putting up an identical-looking page, A’. This page had a collision not with Lurvy’s original page (A), but instead with a page she had created (B’). The beauty of this approach is that she could use an off-the-shelf tool (confoo.pl) to create the pages. But, she had to make the switcheroo before the script ran its first time. That way, the script would be measuring the MD5 hash of a page she had forged (A’) that looked good to Lurvy’s eyes (in a javascript-enabled browser), but had a collision with her anti-advertising page (B’).

4) How should Lurvy's script have functioned to improve its ability to detect the kinds of alterations made by Charlotte?

Many of you hit the right answer here. Lurvy should have used multiple hash algorithms, such as MD5, SHA1, Whirlpool, and RIPEMD-160. With current hash collision technologies, it might be possible to fool one of these (especially MD5, and possibly SHA1), but it is very difficult indeed to fool all of them at the same time. Additionally, Lurvy should have calculated hashes for the entire web site, not just the source of but a single page on the site. He should have looked at other HTML, images, and other elements of the page, too.

My favorite tool for calculating multiple types of hashes of individual files, or even recursively going through entire directories and subdirectories, is md5deep, which runs on Windows, Linux, Mac OS X, and most other UNIXes.

5) What was Charlotte's proposal to Lurvy for saving Wilbur?

In her web site alteration, Charlotte left a hint for Lurvy, saying that the Digital Invisible Ink Toolkit (DIIT) should be used for something. This stego tool can embed hidden information inside of images. I chose this tool for many reasons -- because it is highly useful, well-written, and very reliable. Beyond that, the clincher for me choosing this tool was the fact that it is Java based, meaning that you can run it on any operating system with a Java Runtime Environment. Most Stego tools are either Windows EXEs or Linux binaries. I wanted a solution that you could run on any OS with minimal requirements of installing other software, so I (and Charlotte) chose DIIT.

But, which image contained the hidden message, embedded using DIIT? Well, all of the images on the websites were JPG, except for one, which was a PNG. Also, the name of that PNG page was `counterhackreloadedsteg.png`. See the word `steg` in there? Pretty big hint, no? And, that image file size is much bigger than any of the other images on any of the pages in the challenge. And finally, I wanted you to focus on this image, which contained a blatant advertisement for my book. I've found that some of my challenge-based ads in the past (such as the Base64 encoding of a plug for my book in the Empire Hacks Back challenge) have gone completely unnoticed by anyone working on the challenge. That's a mistake I'm trying to avoid by making these things more blatant.

So, there is something stegographically hidden inside of that PNG image, and DIIT can help you retrieve it. But, DIIT has dozens of config options; plus it requires a password. Which config options should you use? Why, the defaults, of course. And, for the password, you needed to look at Charlotte's defaced page. There were some red letters on the page, which, when put together, spell BAARAMEWE. What the heck is that? Well, for those fans of pig movies out there, these letters are the words Baa Ram Ewe, the secret password used by Babe the Pig in his movie to alert the sheep that he was working with them. This secret password had been handed down from generation to generation of sheep, and was entrusted to Babe under dire circumstances.

And, for those 24 TV show fans in the house, did you notice that Jack Bauer's father in Season 6 bears a striking resemblance to the farmer in the Babe movie? This whole thing smacks of a nefarious plot. Think about it. Charlotte is the best friend of Wilbur, a famous pig just like Babe. Babe knows the farmer, who happens to be the father of Jack Bauer. Jack Bauer had to have his death faked under almost identical circumstances to Charlotte herself. David Palmer knew about both plots. Thus, we have connected Charlotte to Wilbur to Babe to the Farmer to Jack Bauer to President David Palmer back to Charlotte herself, in a very tangled web of deceit and destruction. And, as proof of all of these ties, we have the actor James Cromwell. Coincidence? I don't think so. And, the same actor is going to be in Spider-Man 3 to be released in this Summer. Spider? Webs? Oh, yes; there is something going on here, my friends. Something BIG.

In any event, to use this baaramewe password against the PNG image with DIIT, you'd have to convert it to all lower case. If you do this, DIIT extracts a Microsoft Word file from the image. This file contains a proposal to Lurvy from Charlotte about how to make some money for the farm without sacrificing Wilbur. Specifically, the hidden message said:

`Charlotte's Proposal`

Dear Mr. Lurvy,

Now that I have got your attention, I have a proposal for you. You are obviously a bright businessman, trying to make some money on the sale of Wilbur. But, surely you must recognize the fleeting nature of that one-time sale. I propose to you a better business model, one that can keep this farm profitable for years to come.

Employ me, Charlotte the Spider, as a web site designer, contracting my services out for \$150 per hour. I will charge you only \$ 50 per hour. Thus, working only 40 hours per week, I can bring in more cash for you every single week than the one-time sale of Wilbur. If you are interested in my offer, please send e-mail to `charlotte@counterhack.net`, with the subject: CHARLOTTE'S PROPOSAL.

Yours truly,

Charlotte”

And, that, my friends, was how I approached the challenge myself.

Now, without further adieu, let’s discuss the winners.

In the category of Best Technical Answer… envelope, please. And the winner is:

Gregory Fleischer. In his answer, located here, Gregory nails all of the technical issues quite nicely, and in a brief fashion (Brevity is cool, especially when backed up by major technical kung-fu action). And, then, at the bottom of his answers, he points out some limitations of the confoo tool, and provides a new version that he wrote in Ruby, called confoonu.rb. This new version is very nice, and implements a cleaner attack using chosen identical prefixes before the X and Y colliding blocks (a technique referenced by Gregory here). It makes the resulting collided pages look a lot cleaner than the ones created by the original confoo. Bottom line: if you view the source of the colliding pages, it looks pretty. He also handles embedded relative links better than the original confoo, so his pages display cleanly in IE7, a major accomplishment. So, he nailed all of the technical answers, and released an improved version of confoo…. For that, he wins. Very nice work, Gregory!

Honorable mentions go to:

-

Jon Mark Allen

-

Balazs Attila-Mihaly

-

Ronald Reed

-

Brien Christesen

-

John Matusiak

-

Colin Cashin

-

Candid Wuest

-

Jim Halfpenny

-

Carlos Garcia

-

Shawn Lee

-

Dawn Isabel

-

Hal Coghill

-

Michael Spahn and Bernd Jäger: These gents got really deep into the DIIT decryption, and found some anomalies of the tool, getting false positives on the decryption with passwords other than baaramewe. They also thought I was a bot, responding to their e-mails. All in all, it was an interesting and fun ride watching these guys work!

In the category of Best Creative Answer, the winner is:

Kevin Bong. This answer is so amazing! As envisioned by Kevin, the whole plot was a mind-fake by Lurvy to entrap Charlotte. But, it doesn't end there. Kevin weaves a web of intrigue involving references to Men in Black and elsewhere. His closing line (and image) is priceless. Dude… that was awesome work! My hat’s off to you.

Honorable mentions go to:

Andrew Laman, who did a wonderful technical answer, and also had very amusing creative flair in his response. He even incorporated the songs from the 1970’s version of the Charlotte cartoon.

Adam Sewell and Mary Hill, who even included their own Fibonacci rat speech patterns in their result. It was quite entertaining!

In the random draw category, the winner is:

Hal Coghill. The pseudo-random number generator in my calculator picked out Hal’s entry for a win in this category. He also qualified for Technical Honorable mention, and had a very interesting take on some of the numbers in the story. Quoting Hal, “The bid amounts of \$1,618.03 and \$2718.28 if changed in to latitude and longitude coordinate of hours, minutes, and seconds give a location in Sudan. This is probably the base of the terrorist group that shot the Chinese diplomat.” Whoa… dude… You’ve got quite a vivid imagination. Next thing you know, you’ll be blabbering on about the relationship of Charlotte, Jack Bauer, Babe the pig, and Spider-Man.

And to everyone, thank you all for your incredible insights in working on this challenge. I’m amazed at the abilities

of a large number of you. Competition was fierce, so don't be disappointed if you did not win. Every one of those honorable mentions was fantastic. You don't get an honorable mention unless you did some really really good work. So, congrats to all of the winners, as well as those who received honorable mentions.

In just a few days, we'll be posting our next challenge, created by the inimitable Tom Liston, whose story will be called "Microsoft Office Space".

Thanks again—

--Ed Skoudis.

* Yes, I know.

- All trademarks, images and rights reserved to their original owners.
- Graphics used include the movie image (Paramount Home Video) and a 3D map of the world wide web.