Googling Up Passwords

By *Scott Granneman* Mar 09 2004 12:58PM PT

In my last column, I provided a checklist for Windows users that would help them secure their computers. I created that checklist because it has become increasingly and painfully obvious to me that most home users -- and most small businesses and organizations -- have substandard security practices in place, if they have any at all. The checklist was designed to help secure things on the perimeters: on client computers and at the edges of home and business networks. This week, I want to talk about servers.

Specifically, let's talk about the stuff that people are serving without realizing it. Security pros have known about this problem for years, but most computers users still have no idea that they may be revealing far more to the world than they would want. In fact, it wouldn't be far from the truth to say that Google is in many ways the most useful tool available to the bad guys, and the most dangerous Web site on the Internet for many, many thousands of individuals and organizations.

I'm not putting down Google. Far from it: it's a great search engine, and I use it all the time. I couldn't do my many jobs without Google, so I've spent some time learning how to maximize its value, how to find exactly what I want, how to plumb its depths to find just the right nugget of information that I need. In the same way that Google can be used for good, though, it can also be used by malevolent individuals to root out vulnerabilities, discover passwords and other sensitive data, and in general find out way more about systems than they need to know. And, of course, Google's not the only game in town -- but it is certainly the biggest, the most widely-used, and in many ways the easiest to use.

Throwing Back the Curtain

Most people just head to Google, type in the words they're looking for, and hit Google Search. Some more knowledgeable folks know that they put quotation marks around phrases, or put a "+" in front of required words or a "-" in front of words that should not appear, or even use Boolean search terms like AND, OR, and NOT. Greater Google aficionados know about Google's Advanced Search page, where you get really specific.

The page that Google provides for its Advanced Search is nice, and it's certainly easy and full of necessary tips, but if you really want to master all the tricks that Google offers the dedicated searcher, you need to learn at least some of what is detailed on the Google Advanced Search Operators page. For instance, let's say you just type the word "budget" into a Google search box, without the quotation marks. You're going to get over 11,000,000 hits, so many that it would take a tremendously long time to find anything troublesome from a security perspective.

Now try that same search, but include the search operator "filetype" along with it. Using the filetype operator, you can specify the kind of file you're looking for. Google's Advanced Search page lists several common formats, including Microsoft Word, Microsoft Excel, and Adobe Acrobat PDF, but you actually search for far more than those. Let's change our search from just "budget" to "budget filetype:xls" (again without the quotes; in fact, just ignore the quotation marks unless I mention otherwise) and see what we get.

Hmmm ... now we're down to 63,000 hits. Still an overwhelming number, but if you start looking through the first couple of pages, you'll notice some items of interest if you were an attacker looking for information you shouldn't have. Let's add another operator into the mix.

The "site" operator allows you to narrow down your results to a particular subdomain, a second-level domain, or even a top-level domain. For instance, if you wanted to find out what Google has indexed at SecurityFocus on the topic of password cracking, try this search: "site:www.securityfocus.com password cracking", which gives you 449 results. I often use this trick even when a site provides its own search engine, as Google's index is often far better than the search that many sites include.

Let's try our search, but stick to the .edu top-level domain, so we're looking for "budget filetype:xls site:edu". 15,200 hits. Not bad. Things are starting to look very interesting.

Let's introduce another tool into your toolbox: the ability to look only on pages that use a certain word or words in their title by incorporating the "intitle" operator into your search. At SecurityFocus, this query would narrow our results list down to only 5, an incredible tightening of our search: "site:www.securityfocus.com intitle:password cracking" (note that "password" is the only word that must be in the title; "cracking" should appear on the page as a search term, but not in the title, since I didn't place "intitle:" prior to it).

Bad guys know about the "intitle" operator, but they know something else that makes it even more powerful. Often Web servers are left configured to list the contents of directories if there is no default Web page in those directories; on top of that, those directories often contain lots of stuff that the Web site owners don't actually want to be on the Web. That makes such directory lists prime targets for snoopers. The title of these directory listings almost always start with "Index of", so let's try a new query that I guarantee will generate results that should make you sit up and worry: "intitle:"index of" site:edu password". 2,940 results, and many, if not most, would be completely useless to a potential attacker. Many, however, would yield passwords in plain text, while others could be cracked using common tools like Crack and John the Ripper.

There are other operators, but these should be enough to make the picture clear. Once you start to think about it, the potentially troublesome words and phrases that can be searched for and leveraged should begin to multiply in your mind: passwd. htpasswd. accounts. users.pwd. web_store.cgi. finances. admin. secret. fpadmin.htm. credit card. ssn. And so on. Heck, even "robots.txt" would be useful: after all, if someone doesn't want search engines to find the stuff listed in robots.txt, that stuff could very well be worth a look. Remember, robots.txt just indicates that the Web site doesn't want search engines to index the files and folders listed in robots.txt; nothing inherently stops users from accessing that content once they know it exists.

A couple of Web sites have even sprung up dedicated to listing words and phrases that reveal sensitive information and vulnerabilities. My favorite of these, Googledorks, is a treasure trove of ideas for the budding attacker. As a protective countermeasure, all security pros should visit this site and try out some of the suggestions on the sites that they oversee or with whom they consult. With a little elbow grease, some Perl, and the Google Web API, you could write scripts that would automate the process and generate some nice reports that you could show to your clients. Of course, so could the bad guys ... except I don't think your clients will ever see those reports, just the end results.

Even the Google cache can aid in exposing holes in systems. Couple the operators outlined above with Google's cache, which can provide you with a look at files that have changed or been removed, and attackers have an incredibly powerful tool at their disposal.

Responses

As I said at the beginning of this column, the fact that it is actually quite easy to find dangerous information

using just a search engine and some intelligent guesses is not exactly news to people who think about security professionally. But I'm afraid that there are many uneducated folks putting content onto Web servers that they think is hidden to the world, when it is in reality anything but.

We have two seemingly opposite problems at work here: simplicity and complexity. On the one hand, it has become very easy for non-technical users to post content onto Web servers, sometimes without realizing that they're in fact placing that content on a Web server. It has even become easier to Web-enable databases, which has led in one case to the exposure of a database containing the records of a medical college's patients (and by the way, the search terms discussed in that article are still very much active at Google, one year later).

Even when people do understand that their content is about to go onto the Web, many do not fully think through what they're about to post. They don't examine that content in light of a few simple questions: How could this information be used against me? Or my organization? And should this even go on the Web in the first place?

Well, of course ordinary users don't think to ask these questions! They're just interested in getting their content out there, and most of the time are just pleased as punch that they could publish on the Web in the first place. Critically examining that content for security vulnerabilities is not something they've been trained to do.

On the other side of the coin we have complexity. For all the ease that has come about in the past several years, no matter how simple it has become for Bob in Marketing to publish the company's public sales figures online, the fact remains that we're dealing with complex systems that have many, many points of potential failure. That knowledge scares the hell out of the people who live security, while Bob goes blithely on successfully publishing the company's public sales figures ... and accidentally publishing the spreadsheet containing the company's top customers, complete with contact info, sales figures, and notes about who the salespeople think are good for a few thousand more this year.

For instance, FrontPage is touted by Microsoft as an extremely simple-to-use Web publishing solution that enables users to "move files easily between local and remote locations and publish in both directions". Unfortunately for those average Joes who buy into the hype, FrontPage is still a very complicated program that can easily expose passwords and other sensitive data if it is not administered correctly. Don't believe me? Just search Google for "_vti_pvt password intitle:index.of" and take a look at what you find.

FrontPage is not the only offender, but it is certainly an easy one to find in abundance on our favorite search engine. Now think about all the other programs out there that people are using every day. Personal Web servers that come with operating systems. Turnkey shopping cart software. Web-enabled Access databases. The list goes on and on. Take a moment and start to think about the organizations you oversee. See the list of potential problems tumble off into infinity. Oy.

Sure, it's possible for the folks creating Web content to tell Google and other search engines not to index that content. O'Reilly's Web site has a marvelous short piece titled "Removing Your Materials From Google" that should be required reading for anyone who even thinks about putting anything on or even near a Web server. Of course, as I mentioned above, relying on robots.txt to protect sensitive content is a bit like putting a sign up saying "Please ignore the expensive jewels hidden inside this shack". But at least it will get folks thinking.

And really, that's what it comes down to: we have to get folks thinking. Sure, those of us responsible for security can try to shut everything down and turn everything off that could pose a threat -- and we should, within reason. But those pesky users are going to do their job: use the systems we provide them, and some we don't provide. We need to help them understand the threats that any Web-enabled technology can provide. Print out this column and hand it out. Show them how easy it is to find sensitive content online. Talk to them about appropriate and inappropriate content. Try to get them on your side so they trust you and come to you

with requests for help beforehand instead of coming to you after the fact, when it's too late and the toothpaste is out of the tube. Finally, realize that humans have an innate need to communicate and will seize on any tool to do so, and if that means talking to your users and setting up a wiki or bulletin board or other collaborate tool, then do so.

Google and other search tools have made the world available to us all, if we just know what to ask for. It's our job as security pros to help make the folks we work and interact with aware of that fact, in all of its far-reaching ramifications.

Scott Granneman is a senior consultant for Bryan Consulting Inc. in St. Louis. He specializes in Internet Services and developing Web applications for corporate, educational, and institutional clients.

Copyright © 1999-2004 SecurityFocus