



Practical Security Tips for a Wacky World

By Branden R. Williams – ISSA member, North Texas, USA chapter

Have you ever wanted to see if sensitive data your company protects exists outside of designated areas? Maybe you are looking for Personally Identifiable Information (PII), intellectual property, or cardholder data that might be sitting around in flat files. I suggest turning to Grep,¹ a GNU searching tool that is included on most Unix-based operating systems (and there are MS ports)!

Our customer had no idea they were receiving this information, and yet they would likely be liable for its disclosure if a breach occurred.

Grep can use the power of regular expressions to quickly search for patterns in files. The results obtained will help you triage data leakage that may occur through the normal course of business.

I was recently working with a customer who found additional consumer information in batch files they were receiving from a financial institution. I am not talking about the consumer's hair color or meal preference. I am talking about their social security numbers. Our customer had no idea they were receiving this information, and yet they would likely be liable for its disclosure if a breach occurred.

The method described below is not fool-proof; however, it will find the majority of files that would contain sensitive data.

It will not reliably (without additional tweaking) find information encoded inside binaries or database files, but you can use it to look through compressed files² and flat files.

Let's say you wanted to take a cursory look at data that could contain social security numbers. Your grep command would look like:³

```
grep -r1 "[[:digit:]]{3}\[-,[:space:]]\?[:digit:]]{2}\[-,[:space:]]\?[:digit:]]{4}" / > files-to-triage
```

In this command, we are recursively (-r) looking for any file that has nine digits in a row, with or without delimiters, and listing those filenames (-l) into a file called *files-to-triage*. After doing some test runs with this, I discovered that this will match quite a few files that do not have any of this data, but I think I could filter those out pretty quickly.

What if you wanted to look for stray cardholder data? To find most variations of valid cardholder data (apologies if I missed any) that would occur in a solid string of numbers or blocks of four separated by dashes or spaces, you would type:

```
grep -r1 "\(\(4[[:digit:]]{3}\)\|\(5[1-5][[:digit:]]{2}\)\|\(6011\)\)\[-,[:space:]]\?[:digit:]]{4}\[-,[:space:]]\?[:digit:]]{4}\[-,[:space:]]\?[:digit:]]{4}\|3[4,7][[:digit:]]{13}\|3[0,6,8][[:digit:]]{12}\)" / | mail -s "Sensitive Data Report for `uname -a` sensitive_data_reporting@company.com"
```

This is what it looks like to write a regular expression that will search for most known types of credit card data. Regular expressions look like a work of art when viewed by the trained eye, but like hieroglyphics for the rest of us.

But what is that new fancy piece on the end? Say you wanted to run this regularly and email it to a report-gathering mailbox. Provided your server has the mail program installed and its communication with a mail server is not blocked by firewall rules, this may be acceptable.

The automation can get pretty sophisticated from here. You could enhance the regular expression more or further parse the output to remove false positives, have the information dropped into a database or aggregate report on a daily basis with scripts, or simply just run it on a schedule from *cron*.

Now keep in mind, this is not a fool-proof approach, nor the most efficient way to search for these types of data. There are several commercial packages on the market today to choose from that will do what you see above, and much more. But if you are looking for a starting point to see if you have a problem, using open source tools like Grep can be a cost-efficient way to see how deep the hole really goes.

About the Author

Branden R. Williams, CISSP, CISM, is currently the Director of the PCI Consulting Practice at VeriSign and regularly consults with top global retailers, financial institutions, and multinationals. He can be reached at bwilliams@verisign.com or at <http://www.brandenwilliams.com>.

¹ <http://www.gnu.org/software/grep>

² To do this, you need to use "zgrep."

³ Note: you may have slight syntax differences in your regular expressions. Non-maintained versions of operating systems need not apply.