If You Give a Dev a Library...

Letters: Josh More

Pictures: Shea Bartel

Sponsored by

EYRA

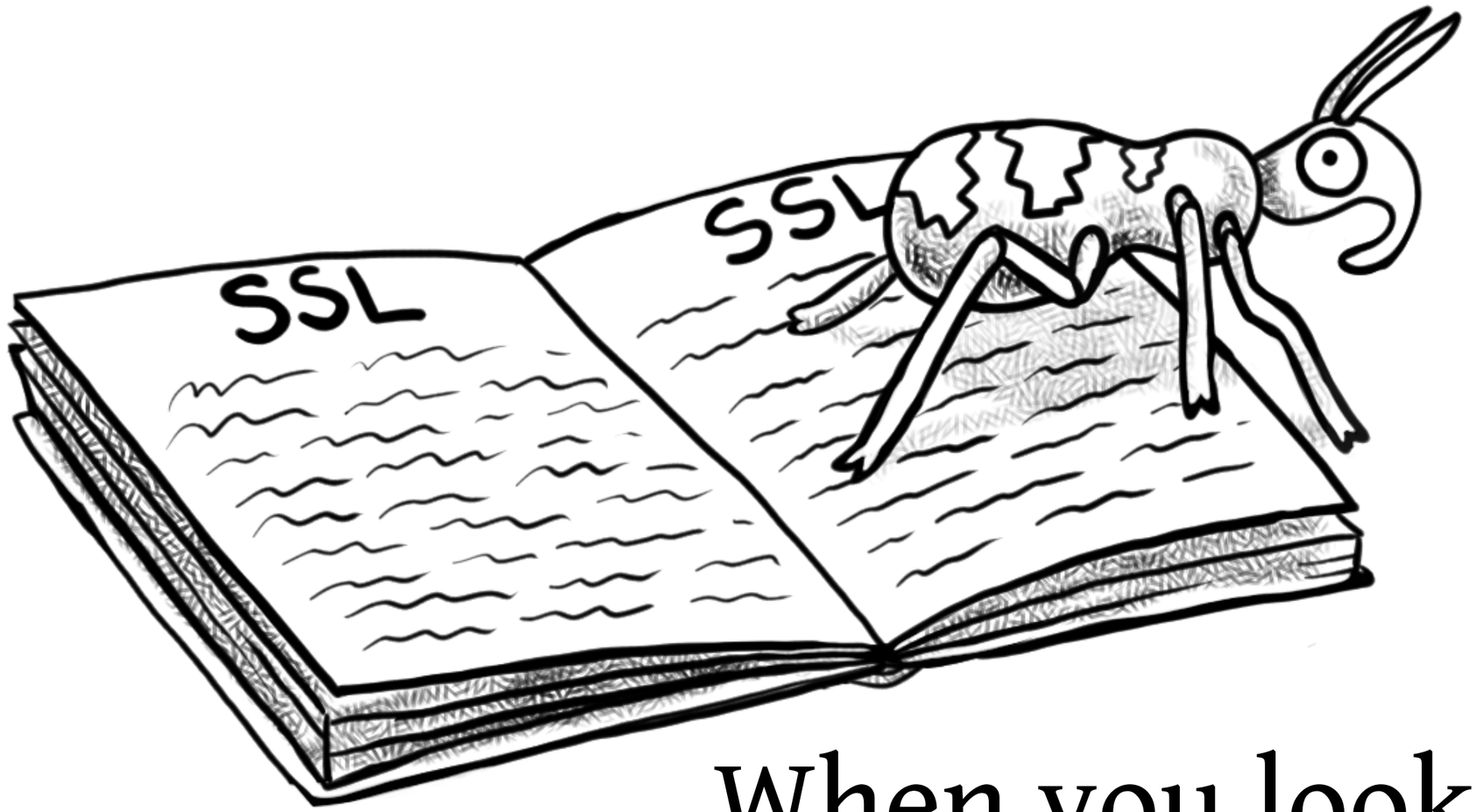
Version 1.0

If you give a dev a library...

NOTE

The OpenSSL library allows programmers to add proven security to their applications quickly and easily.

It'll probably have a bug in it.

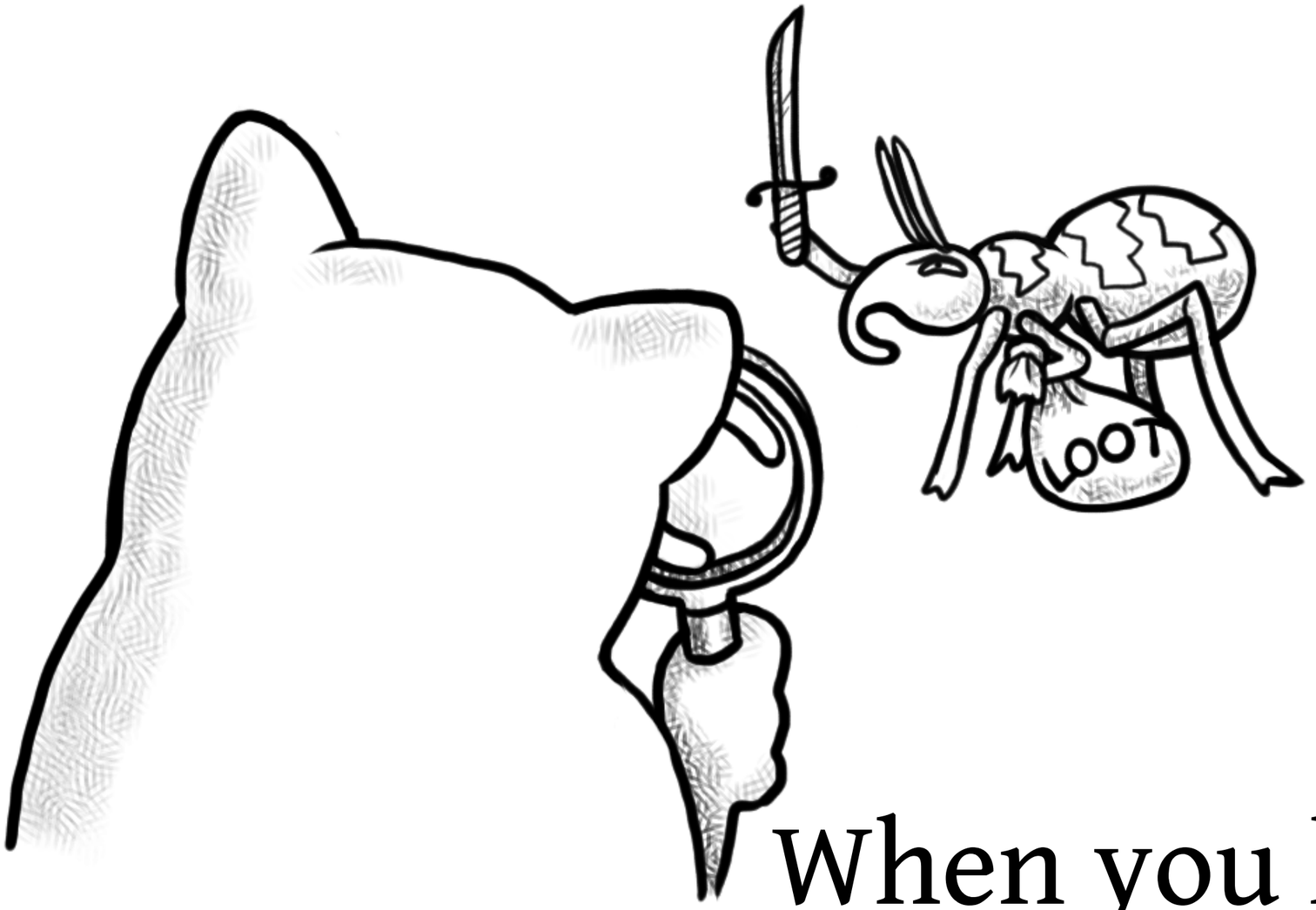


When you look at the bug...

NOTE

Some versions of OpenSSL, alas, have an error. It's fixed now.

You'll see it might be dangerous.

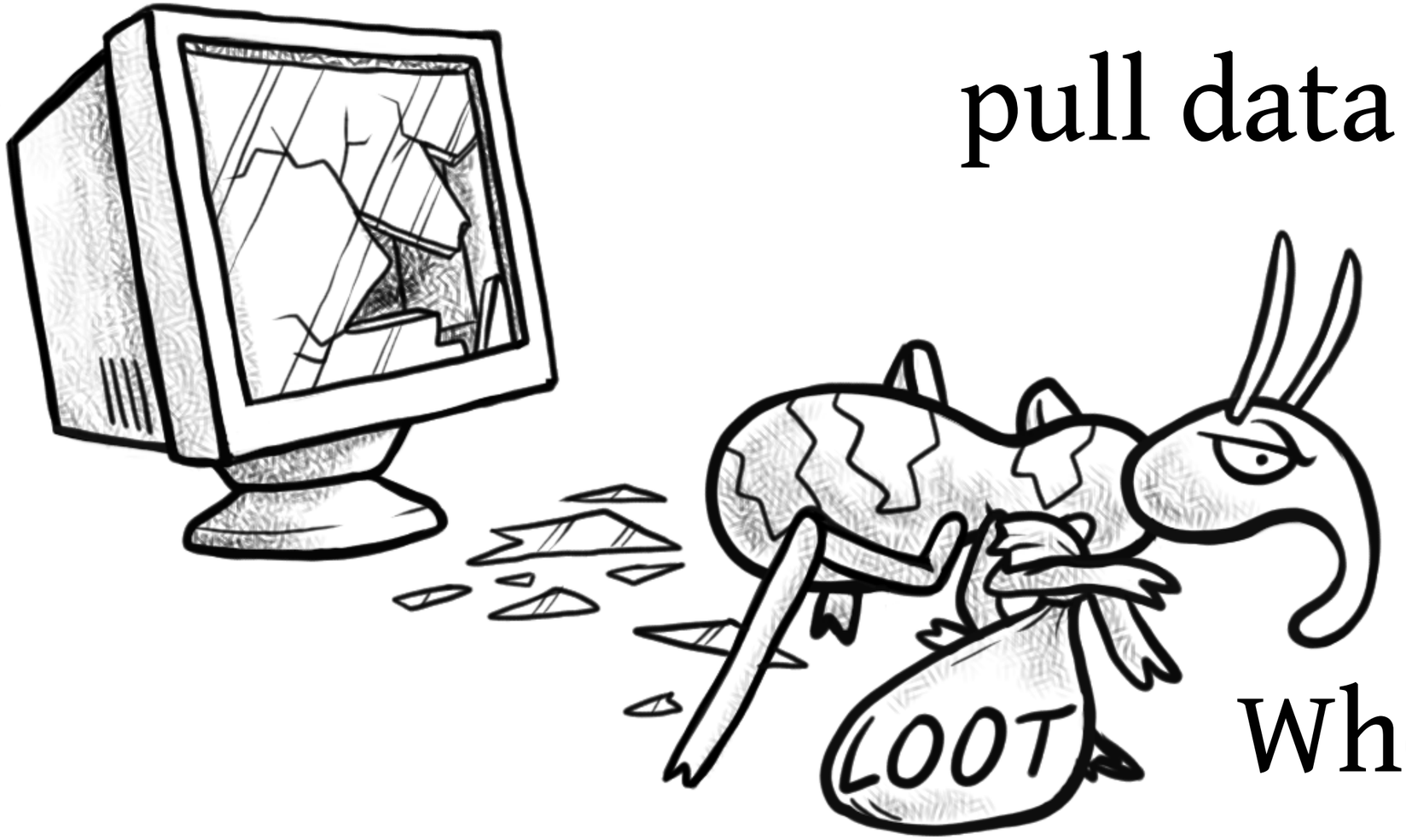


When you look to see why...

NOTE

The problem in the library existed from March 2012 to April 2014. The problem on the Internet will likely be with us forever.

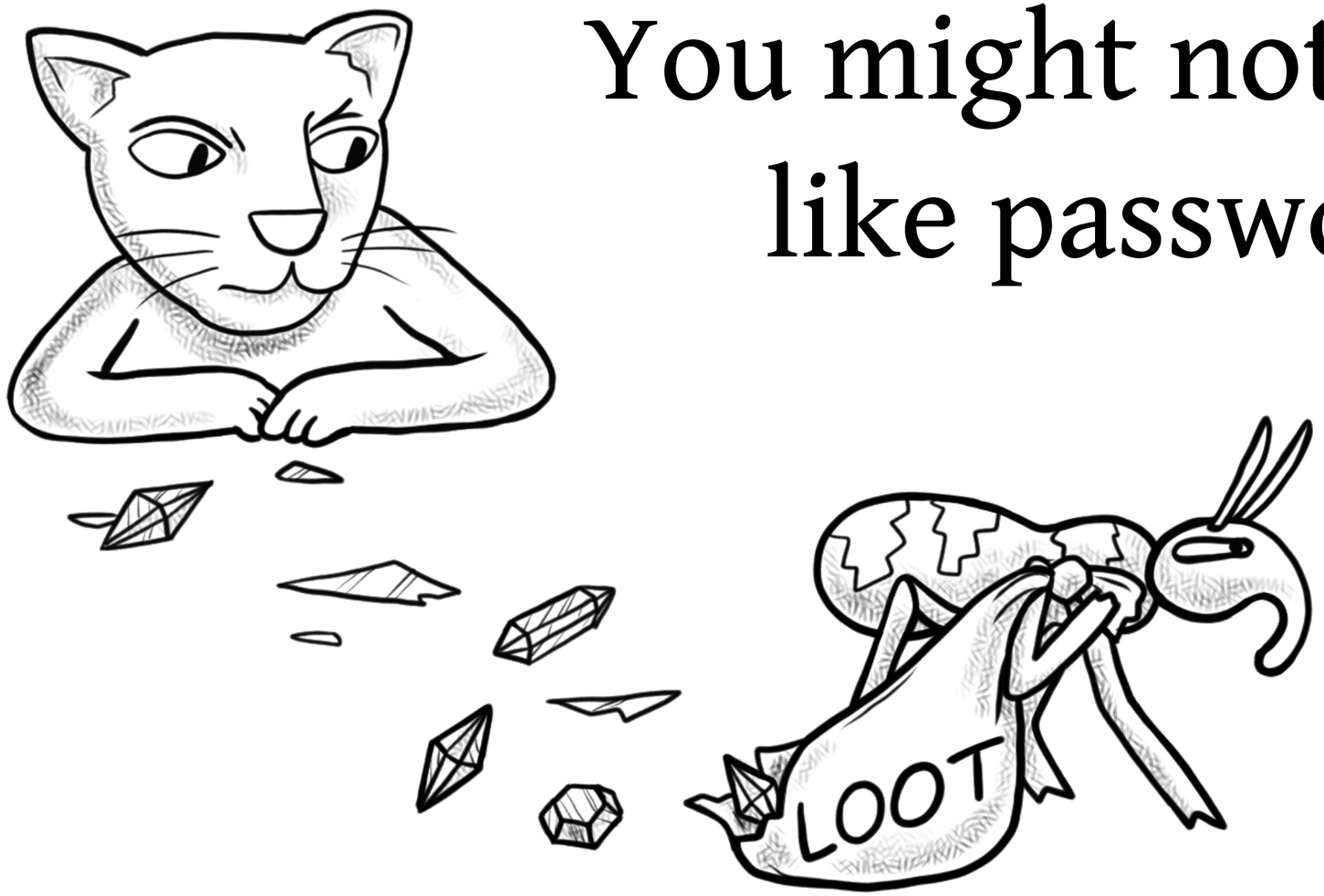
You'll see it lets attackers pull data from "secure" web sites.



When you look at the web sites...

NOTE

When attackers do this, there's no way for us to know, so if data is stolen, we can't know that either. To be safe, we must assume it's been stolen.



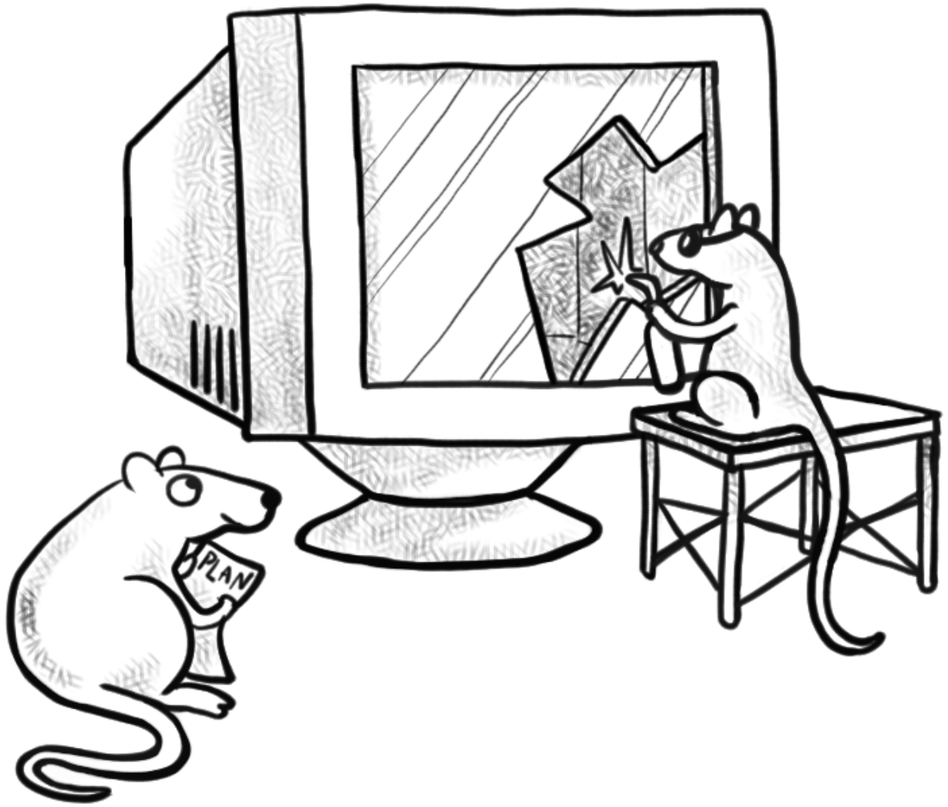
You might notice valuable info, like passwords and keys.

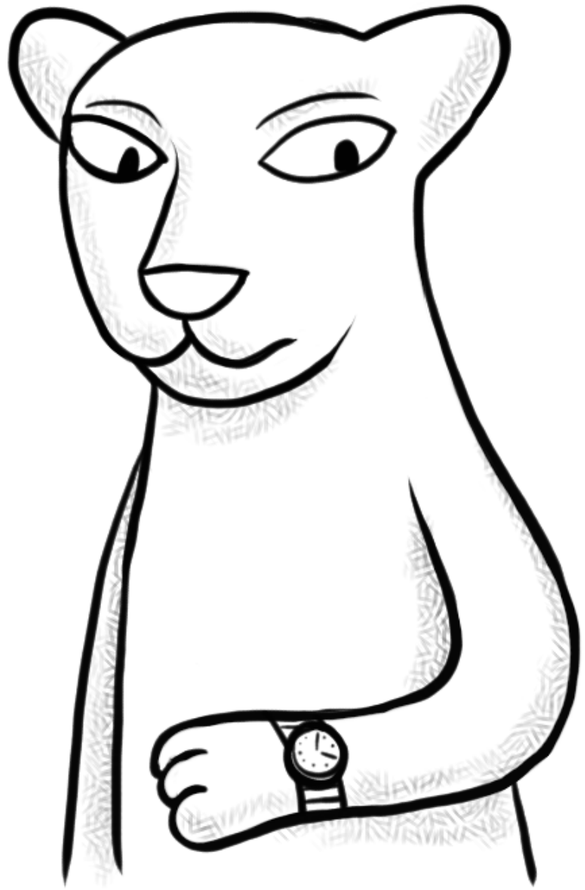
So you'll probably want to change them...

NOTE

Attackers can see passwords, encryption keys and sensitive data. You can change your password.
If you run a server, you can change your certificates.
Changing social security numbers, bank accounts, medical history ... that's a lot harder.

But if the sites aren't fixed, you'll have to change them later.



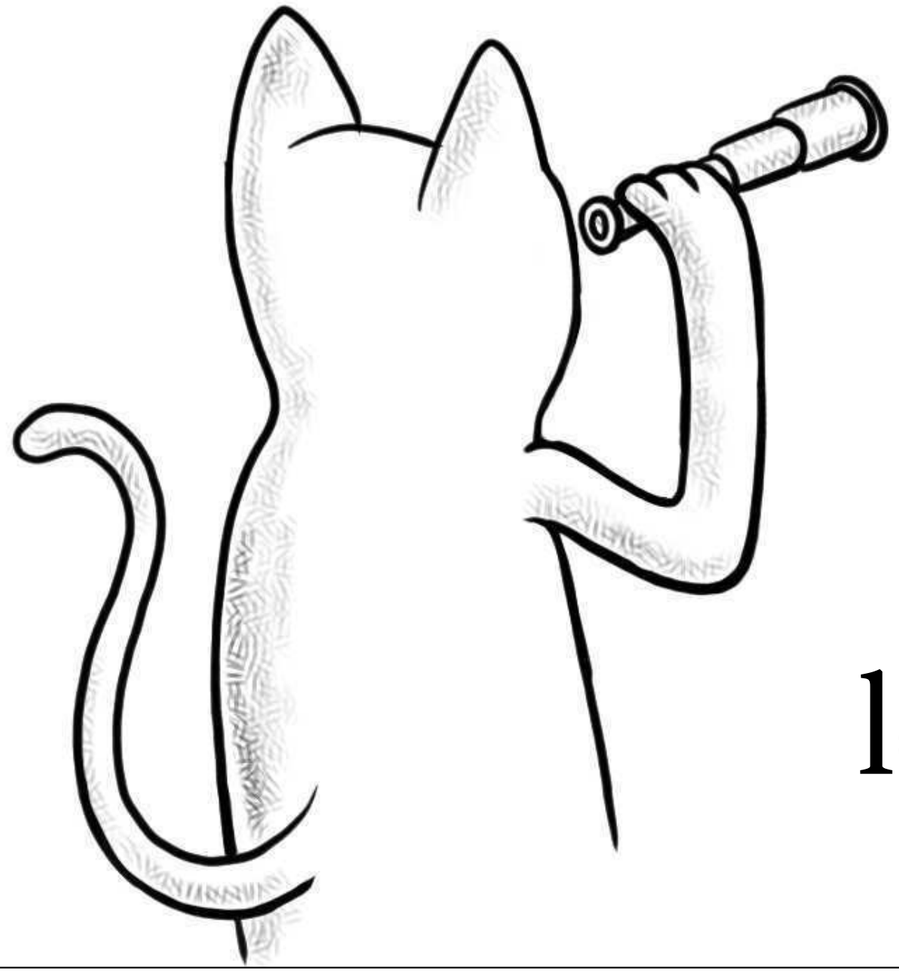


You'll start watching...

NOTE

If you change your info before the system is running the fixed version of OpenSSL, attackers can still get it, so you'll want to wait.

every site on the Internet
to know when it's safe.



You may even end up looking at VPNs as well!

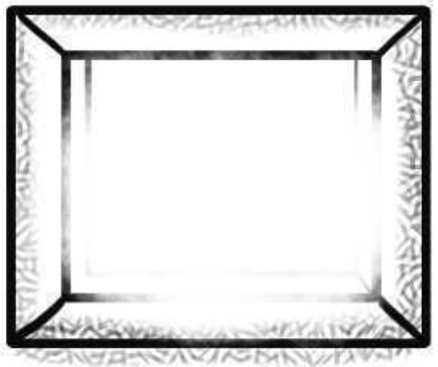
NOTE

Mashable lists affected sites. Zmap lists known-vulnerable sites. Neither of these lists is complete. Test with SSLLabs to be sure.

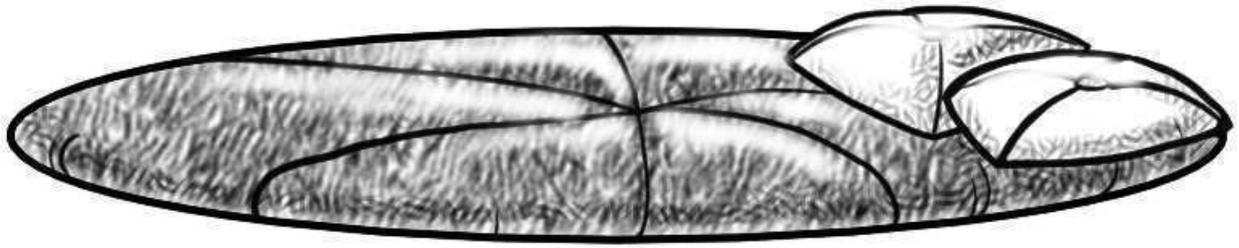
NOTE

Many implementations of OpenVPN are vulnerable.

Once done, you'll want to rest.



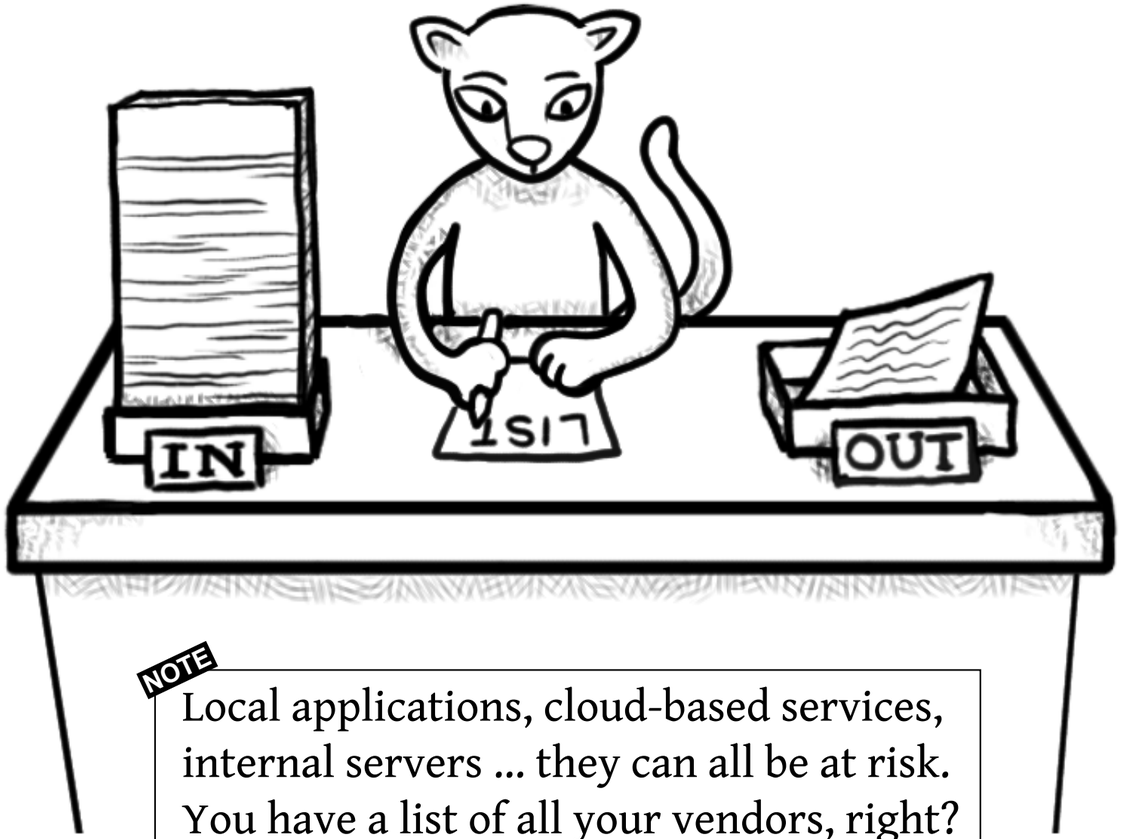
But there's more to do...



NOTE

The Internet, So relaxing.

involving vendor services and software.



NOTE
Local applications, cloud-based services, internal servers ... they can all be at risk. You have a list of all your vendors, right?

NOTE
-sigh-

You'll pull a list, of all your vendors, review each one...

and likely miss a few.

NOTE
Go pull five year's worth of reports from your accounts payable system. You should probably do this at least once per year.

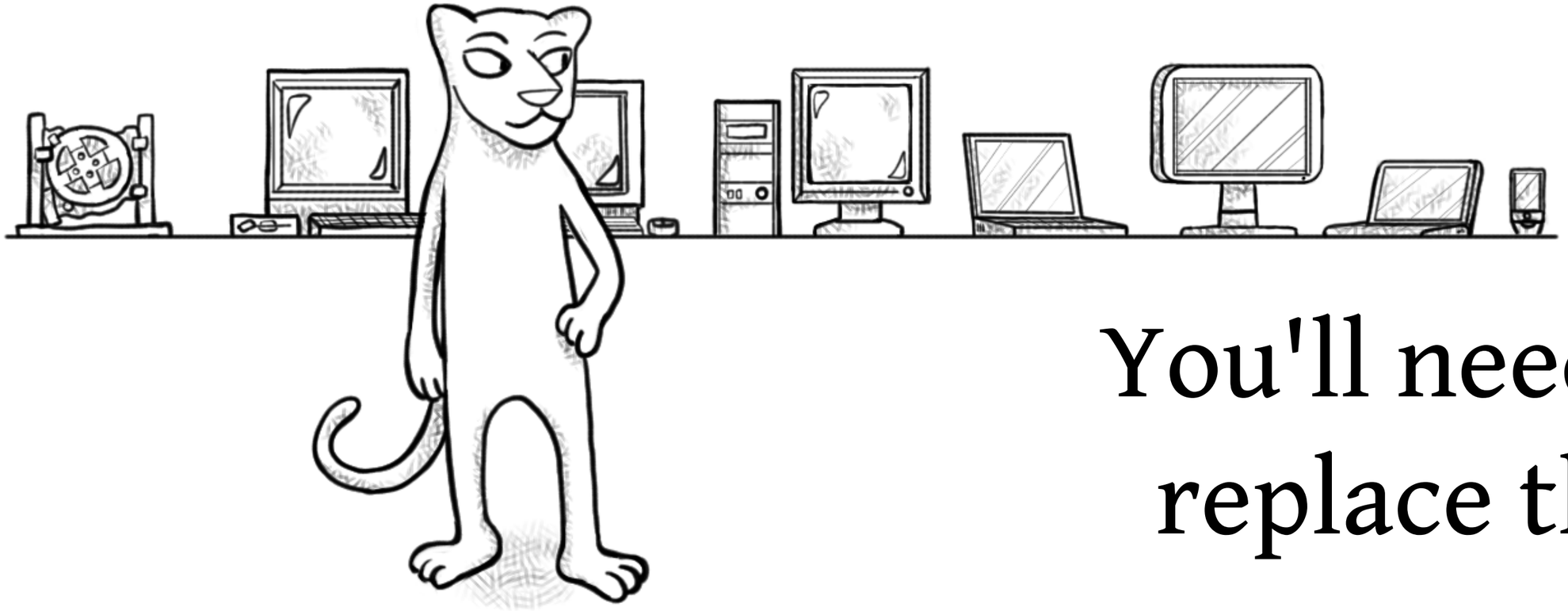
So you'll contact each one and ask if you're safe;

some won't get back to you.

NOTE

You should probably have contact info for each of your vendors added to your vendor list.

When you review the list, you'll see old, unsupported systems.

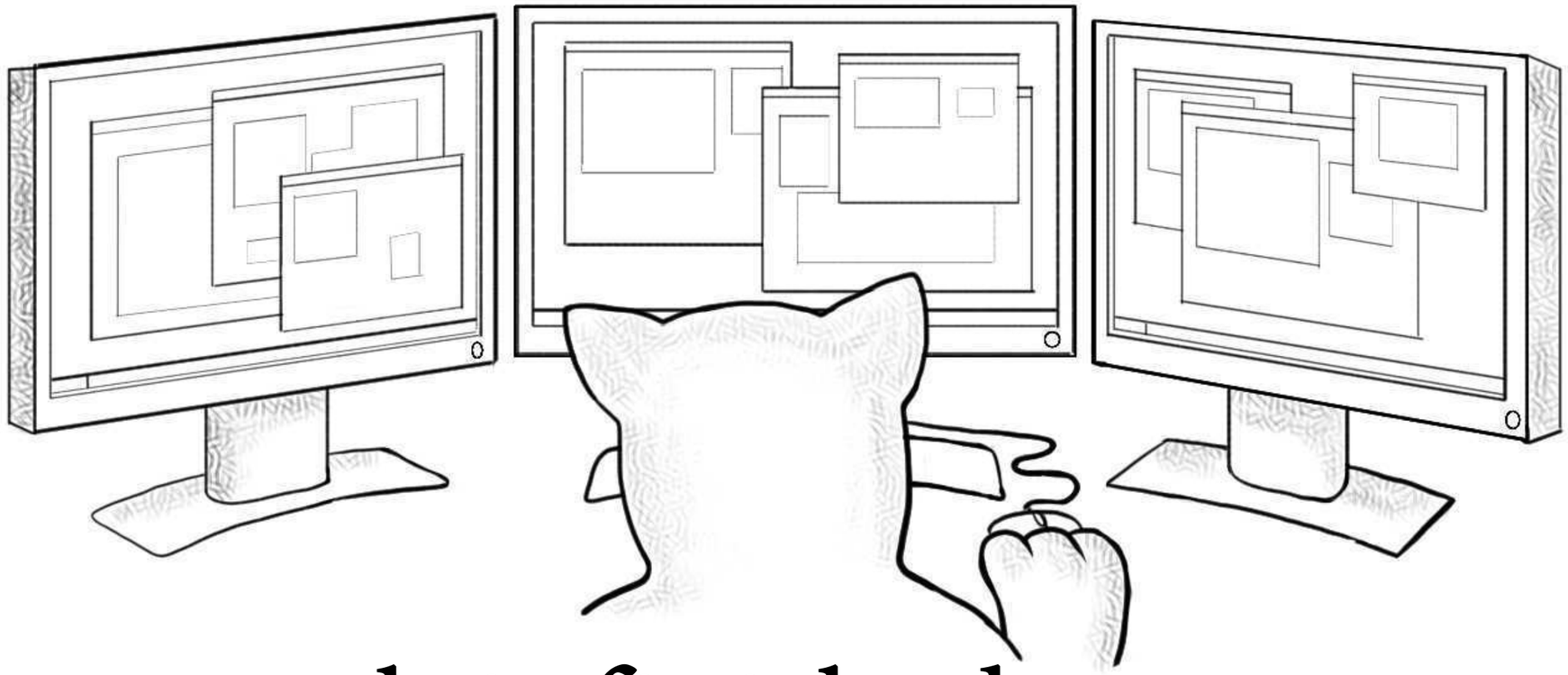


You'll need to replace them...

NOTE

If you have any applications or services in use and you've not worked with your vendor in five years, there's no relationship. This problem is bigger than HeartBleed.

And review your options.

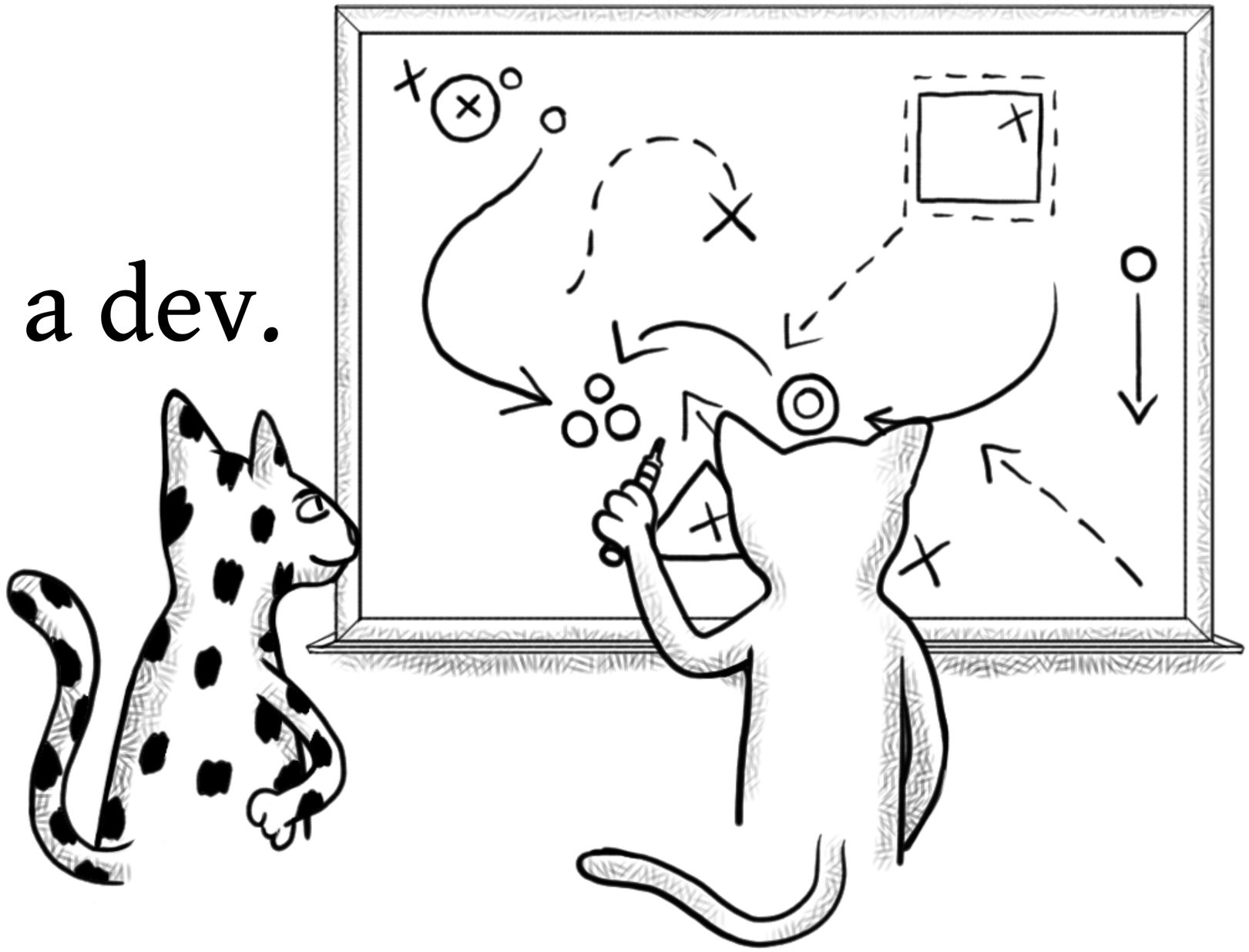


When your research is finished, you may choose to build your own...

NOTE

You should look at commercial and non-commercial, closed and open source and custom-built options.
Shameless plug: I wrote a book on how to do this. (Also available at Amazon and B&N.)

With a dev.

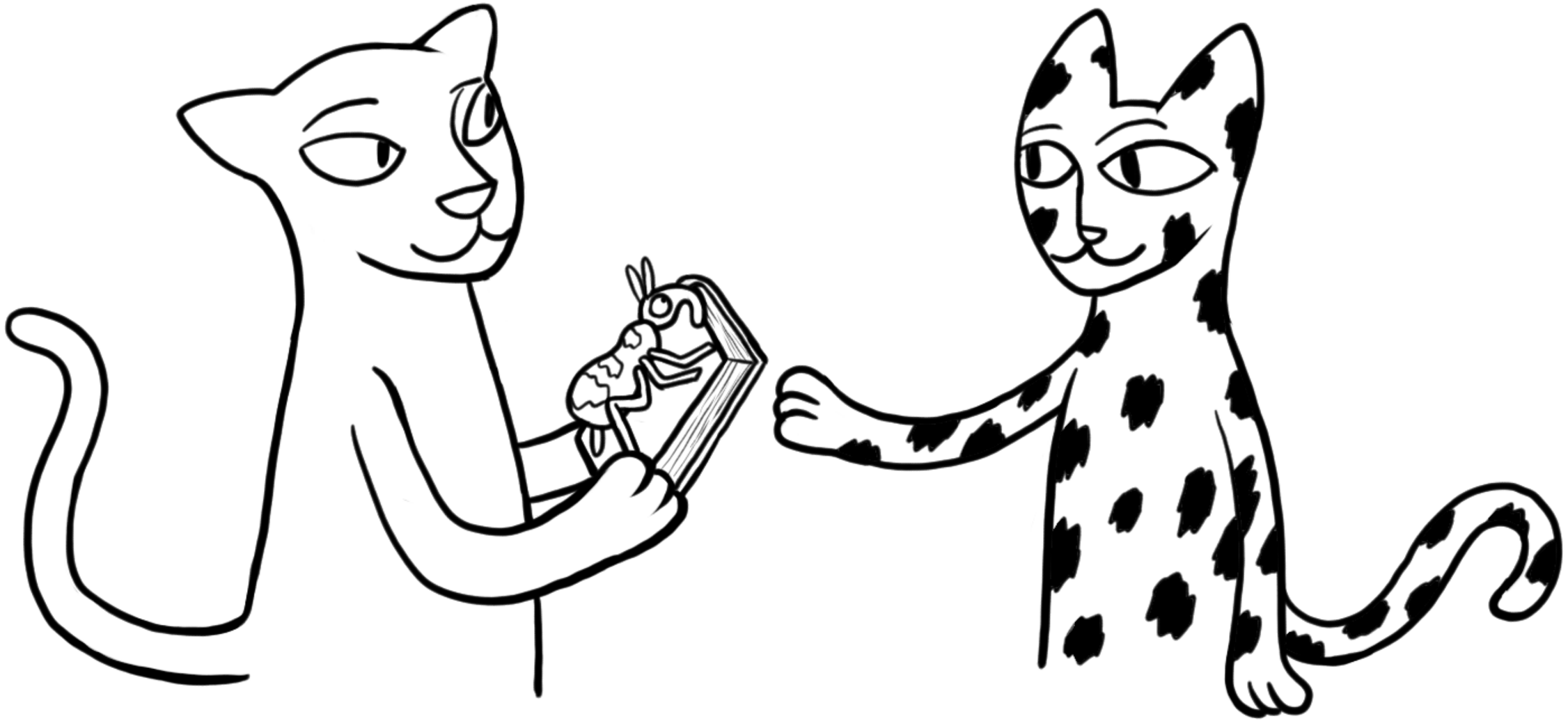


Who'll likely want a library....

NOTE

The best way to get a tool that fits all your needs is to build it.
The best way to get your business model stuck in the past is to have a custom tool that you can no longer maintain.

With a bug in it.



NOTE

Developers are expensive. Libraries help save them time.

A moment of seriousness

This fun and silly e-book is really just saying security is hard. The HeartBleed vulnerability is but one of many. As I write this, HeartBleed is becoming well-known because OpenSSL is widely used. However, there are many similar tools. In a way, we're lucky. HeartBleed's media coverage has many people checking their systems. We won't be as lucky next time.

Security requires thought. “Patch everything” fails as total patching gets progressively harder as environments grow more complex. This approach also fails in situations where the patch provides mere groundwork. Microsoft's KB2269637, KB2719662 and Debian's CVE-2008-0166 are like HeartBleed in that additional work is required after patching to ensure the problem is fixed.

Security is messy. The advice to change passwords and SSL keys is over-simplified. The likelihood of your key being lost is much higher if your server was rebooted around April 7th, 2014. The likelihood of a password loss is higher if you were logged in to a vulnerable site during this time. However if you don't know when your vendor fixed HeartBleed, you don't know when the timeframe ends. Additionally, since we don't know when HeartBleed was first attacked, we don't know when the timeframe starts.

Security is imperfect. You may know when you get compromised, but you can never know that you weren't. How do you walk the fine line between confidence and paranoia? How do you determine which issues to prevent and which to accept? Do you truly believe that so-called “industry best practices” make sense for you and your business? If everyone were to follow them, it makes it easy for the attackers to identify what will and won't work.

Security is a losing game. Attackers are better capitalized, having the time, money and skill to get better every day. We have restrictions - budgets, laws, regulations and standards to follow. We can't always take the time to do things right. The attackers have to win once. We must defend 24 hours a day, 7 days a week and 365(1/4) days a year. It's exhausting.

We need help

Eyra Security is built around this idea. By helping our clients make small, prioritized, steps to improvement, we internalize security into project estimates, operational practices and the general culture.

As you go through life, you start out learning. Many people spend around twenty years doing nothing but learning. Eventually, they start using their knowledge. The more people that move from “learning” to “doing”, the better our society functions, the more our economy grows, and the more our standards of living rise. However, if people don't take a break from doing” to inform the next generation, we'll never improve as quickly as the attackers.

The answer is teaching. Teaching is a force multiplier. The more we help the next generation learn, the more people we eventually have helping us with the “doing” side of things. When you teach, you create resources and, along the way, improve your learning to become increasingly effective. Since security requires analysis and consideration to weigh risks and make the right decisions, the better we get at making decisions, the more secure we become.

If you want to chat about how the process works and how we may help you, please let us know: info@eyrasecurity.com

Thank you for reading my little e-book.

-Josh More
President, Eyra Security

NOTE

Inspired by *If You Give a Mouse a Cookie* by Laura Joffe Numeroff. (Also available at Amazon and B&N.)

EYRA SECURITY