(target: bbuzz 16)

2015-11-23 Household INFOSEC in a Post-Sony era

The attack on Sony led to lots of talk about how they could have defended themselves.

Is your household any better? Have you defence in depth? Is all that protects you anISP's base station hasn't had an upgrade for four years? How often do you upgrade Adobe Flash on your TV? And what is it you really need to worry about in a world of internet-enabled-everything?

I examine the threat model and attack vectors, going beyond "keep your PC up to date" to the new problems: smart TVs, fitness trackers, smartphone apps —and the INFOSEC issues of modern cars. In the process: whether you are publishing so much private data that worrying about laptop security is moot. Finally, it considers whether your github credentials are strategic data to nation states.

After this talk you'll not only want an OSS router, you'll be packet sniffing your TV, encrypting your sensitive data off-site, and, the next time you buy an automobile, making sure the previous owner drove safely and disabled flash.

Takeaway

Steve Loughran is an R&D engineer at Hortonworks, where he works on leading-edge developments within the Hadoop ecosystem. He is the author of Ant in Action, a member of the Apache Software Foundation, an active committer on the Hadoop core projects.

Steve Loughran is an R&D engineer at Hortonworks, where he works on leading-edge developments within the Hadoop ecosystem, including service availability, cloud infrastructure integration, and emerging layers in the Hadoop stack. He is the author of Ant in Action, a member of the Apache Software Foundation, an active committer on the Hadoop core projects..

He lives and works in Bristol, England. Trying to maintain household information secure, —despite the presence of a teenage boy and numerous devices— is a battle he is at least aware he's losing.

Past speaking experience includes: numerous ApacheCon EU talks (2005-2015), Berlin Buzzwords (x3), Strata EU, OSCON EU, Hadoop Summit EU, +others.

Conference committee of Berlin Buzzwords & Hadoop Summit EU

twitter: @steveloughran
slides: http://slideshare.net/steve_l

ORA submission extras

Summary

A tour of the security vulnerabilities of a modern household, what can be done to migitate some, why this is too late in the product pipeline —and why OSS developers have the responsibility to code securely and lock down their development systems.

Takeaway

While we worry about keeping our desktop browsers up to date, the current generation of IoT devices: cars, televisions, fitness trackers are re-implementing the same security problems, and adding the intentional capture and uploading of our personal data. The ubiquity of Open Source offers some defences here —but it means we have to lead the way: writing secure code and making sure we get our own houses in order: literally.

Prerequisites

Basic experience in keeping a desktop computer up to date with security patches, configuring home routers

Motivation can come from having bought a television, motor vehicle, fitness tracker or ever installed an airline checkin application on a smartphone —and an interest in keeping their products and personal data secure and private.

Bristech

Notes

I'm giving a precursor this talk at Berlin Buzzwords in June; what I'm proposing here will be an evolution.

As you can see from set of the images which I'll be picking slides from, this is not a talk about DD-WRT router config, ipsec rules, etc.

https://www.flickr.com/photos/steve_l/albums/72157667789976296

It's going to consider keep a house locked down when there are televisions recording what you watch, your phone is tracking your movements and your heartbeat —
and whether building flash into cars is a good idea (answer: there is no possible valid reason ever).

The fact that cars are shipping with flash enabled means that by the time we consumers get hardware, it is too late. We developers need to start by setting the example: understanding privacy when collecting (or not collecting) data, understanding security in every line of code we right, with the core theme being "assume all external data is malicious".