Road Warrior Disaster Recovery

Secure, Synchronized, and Backed-up

Aaron Poffenberger

akp@hypernote.com

2019-05-18T20:00:00Z

Aaron Poffenberger

- Software developer
- OpenBSD user since ~3.2
- Ersatz Backup Operator

- Overview
    - Motivation
    - Goals
- Disaster Recovery Planning
    - What is a Disaster?
    - Disasters Travellers Face
    - Risks
- Preparation and Operation
    - System Hardening
    - Data Synchronization
    - Backups
- Disaster Recovery
- Preparing to Cross International Borders

Motivation

- That time I nuked the disklabel
- Increasing amount and importance of data on laptops

Goals

- Reliable, fast backups at rest and on the go
- Easy to access and restore while travelling
- Sync `$HOME` with other systems
- Tools available in base or packages
- No compromise on security

What is a Disaster?

For my purposes, a disaster is any event that prevents me from using my laptop or accessing the data I need.

Disasters Travellers Face

- Hardware failure
- Theft
- Confiscation
- User mistakes, ahem, `dd(1)`

Disaster Recovery Planning

Some questions I considered when developing my disaster recovery plan.

Who am I in the world?

- CEO, CTO, CFO
- System Administrator
- Developer
- Journalist, activist, dissident, gadfly

What sensitive data do I have?

- Source code
- Access codes
- Customer data

What access do I have that someone might want?

- Admin/root
- Personal or employer banking details
- Social media accounts
- Customer VPN credentials
- Commit bit to an interesting project

How is that hardware, sensitive data, or access at risk?

- Thieves, fraudsters, et al. looking for a quick buck
- Competitors
- Nation states
- Manufacturers (hardware failure)
- Self ("I left my laptop on the plane.")

My Answers

- I’m a Developer
- Who carries mostly personal data
- With:
  - root access to personal and customer servers
  - Personal banking details
  - Social media accounts
- Whose data is mostly interesting to:
  - Thieves, fraudsters, et al.
  - Maybe to some competitors
  - My own mistakes or hardware failures

Preparation and Operation

With those answers in mind, I began looking at how best to prepare for the inevitable disaster.

System Hardening - BIOS

The first step was hardening the laptop at the BIOS level to provide tamper evidence, and to prevent surreptitious access.

- Bottom cover open warning
- Supervisor password
- Boot OS drive only, require supervisor password to boot USB disks, CD ROMs or PXE

System Hardening - Full-Disk Encryption

Full-disk encryption is the lynchpin of the disaster recovery plan. It provides:

- Peace of mind if the computer goes missing
- Reduces anxiety about throwing out old hard drives or computers

System Hardening - OS

- Set `AllowUsers` in `sshd_config(5)`
- All user-editable config files maintained separately and installed with `rdist(1)`.
- `/usr/sbin/apm -Z` in `crontab(1)` to hibernate daily Because it's hard to hack past full-disk encryption.
- `hotplugd(8)` script to lock the screen on insertion of USB HIDs (human-interface devices) like keyboards and mice.

I know what you’re thinking

“Drink the tinfoil-hat Kool-Aid much?”

Disaster Planning - Two Sides

After a disaster the first question everyone asks is:

```
How do we recover from it?
```

But there are two sides to every disaster. The other question to ask is:

```
What happened to the data on the systems in
the disaster?
```

Recovery is important, but it's just important to ensure no one can benefit from the disaster.

Failing Safely

Make sure family can still get into the computer and password manager if something catasrophic happens. The goal is to recover from disaster and keep the bad guys out.

Preparation and Operation

With the laptop and OS hardened, I turned to synchronization and backups.

Data Synchronization

Repat after me: “Synchronization is not a backup strategy.”

But still useful for syncing data with other systems, and for getting a current copy of `$HOME` after disaster recovery.

Data Synchronization

For synchronization I chose Unison over `ssh(1)`.

- Custom script in `crontab(1)` to call `unison` every `n` minutes
  - `$HOME` only
- `ssh-keygen(1)` signing certificate
- Add "TrustedUserCAKeys /etc/ssh/ca.pub" to `sshd_config(5)`
- Entry in `ssh_config(5)` to ensure connectivity at home and away
  - Use `ssh` `ProxyJump` to traverse firewall

What's ProxyJump (-J)?

Ever wondered how to connect to your host on the other side of the firewall? **ProxyJump**

```
Setting this option will cause ssh(1) to connect to
the target host by first making a ssh(1) connection
to the specified ProxyJump host and then establishing
a TCP forwarding to the ultimate target from there.
```

Yes, you can make multiple jumps with a comma-delimited list of ProxyJump hosts.

ssh_config

```
Host fw
   HostKeyAlias fw.tld
Match Host fw !exec "host fw"
   HostName <static_ip>

Host homenas
   HostKeyAlias homenas.tld
Match Host homenas !exec "host homenas"
   ProxyJump proxy@fw
```

Backups

Synchronization (and snapshots) are not the same as backups.

C.f.: The Untold Story of NotPetya

Backups

For backups I use `rsync` to copy key directories to remote server with a snapshotting filesystem, and a hard drive I carry with me.

- Custom script in `crontab(1)` to call `rsync` every `n` hours
    - `$HOME`, `/root`, `/var`, `/usr/src`, `/usr/ports`, `/usr/xenocara`
- Uses same entries in `ssh_config(5)`, and signing certificate as above
- `apmd(8)` attach script that checks whether the DUID of inserted media is in the list of known `softraid(4)` crypto disks, attaches the disk, mounts the volume, makes a backup, unmounts, and deletes the volume.

Localhost Security

Yes, I store some passwords in the clear on the box. *E.g.*, `softraid(4)` crypto passwords. It's a single-user system. The files are `chown`'d and `chmod`'d.

If the bad guys are on my computer and can read files in `/etc/ssl/`, I have bigger problems than a few FDE passwords lying around.

Disaster Recovery

Requires some preparation. Very hard to do *post hoc*, but possible, depending on the disaster.

C.f.: That Time I Nuked the Disklabel

Disaster Recovery

- OS Install Drive
- Recent backup (optional, but very encouraged)
    - `rsync`ing 100GB across hotel wifi hurts

Disaster Recovery - OS Install Drive

OpenBSD -*current* - **with** necessary firmware

Could be integrated with backup drive, but not a priority for me.

Recovery Steps

Installing OpenBSD takes < 10 minutes.

Now that we have `openrsync(1)` in base I don't have to run `pkg_add(1)`. I can begin restoring immediately.

And now that we have `sysupgrade(8)`, I can pick-up any changes since my OS install disk was last created.

Finally, once that's complete, I run `unison` to pick-up the latest synchronized files, `et voilà`, I'm done.

Does it Work?

Yes, quite well.

I recently took trip 500 miles from home with nothing but a laptop running Windows and my recovery disks. Came back with a laptop running OpenBSD and all my data.

Preparing to Cross International Borders

International travel poses special, significant risks to travellers. The threat of confiscation of phones, laptops, and other digital devices can give one pause when travelling.

How do we travel safely?

A Good Disaster-Recovery Plan Breeds Confidence

Given the reliability of the system I've crafted for myself, I feel no hesitance about purposely running:

```
dd if=/dev/random of=/dev/rsd0c bs=1M count=3
```

Or carrying Windows restore media with me and reverting the computer to an OS border patrol would recognize and understand.

Other Options

If nuking the disklabel is too scary, consider paring down the amount and scope of data you travel with:

- Can you archive financial records from years past?
- Can you store current financial records on system that doesn't travel?
- Do you need to carry all those passwords while you travel?

Or consider carrying a “dumb terminal” laptop with you and connecting to a remote system.

Conclusion

You have questions, I may have answers

Contact Details

- Aaron Poffenberger
- akp@hypernote.com
- Blog: http://akpoff.com
- Twitter: @akpoff
- bsd.network: @akpoff
- Amateur Radio: KG5DQJ

Slides for this presentation will be posted on my blog and BSDCan

Thanks

- BSDCan, Sponsors, and Volunteers
- OpenBSD
- OpenSSH
- rsync / OpenRsync
- Unison

Support OpenBSD

- http://www.openbsdfoundation.org/

Further Reading

- That Time I Nuked the Disklabel
- The Untold Story of NotPetya

Technologies and Resources

- git-annex
- [KeepassXC]
- Onlykey
- OpenBSD
- OpenSSH
- rsync / OpenRsync
- sshfs
- Syncthing
- Tarsnap
- Unison
- Wikipedia File Synchronization Comparison