ISPConfig Misconfiguration Leads All Web Sites Offline

Original Issue

When visiting a web site on our ISPConfig3-managed server, all sites returned `SSL_ERROR_RX_RECORD_TOO_LONG` to the web browser. We never did discover exactly what caused the error, but the following are the tests we performed and the simple solution we found.

We had a team of three technicians working on the problem. I took the lead, as I was the most knowledgeable of this particular setup.

We had two machines open, each configured the same except for the actual domains. `good.example.org` was functioning perfectly, and `bad.example.org` returned `SSL_ERROR_RX_RECORD_TOO_LONG` when going to a site via a web browser.

- Compared `good.example.org` with `bad.example.org`. They have different domains, but otherwise are pretty much the same.
  - `/etc/apache2/sites-enabled` and `sites-available` appeared comparable
  - `/var/www` on both machines showed a similar structure
  - MySQL appeared to respond similarly on both machines
- Checked SSL certs
  - All were new, i.e., the creation dates were within the life of the certificates
- Restarted Apache2
  - No issue
- Ran command `apachectl -S`
  - Normally, this command returns a long list of all virtual sites (vhosts), but it only returned the main site
- Attempted to reinstall any old certificates with the following command:
  DO NOT run this based on an AI suggestion alone – this command did not fix the original problem, and running it a second time later corrupted a working server:
  `/root/.acme.sh/acme.sh –cron –home /root/.acme.sh`
  - It found that most sites did not need new certificates

We were under a time crunch (management breathing down our necks), so we finally decided to simply recover from a backup made four days earlier. It turned out that caused a few problems, as Marketing had made changes that weren't in the backup.

During the process, major mistakes were made:

- One of the team suggested checking disk space, which I did not do. That very possibly would have allowed a complete recovery (see solution below).
- I accidentally ran the following command on `good.example.org`, which apparently corrupted it:
  DO NOT run this against a server that is already working:
  `/root/.acme.sh/acme.sh –cron –home /root/.acme.sh`
- I relied on AI (GPT-5.4 mini) too much, and not enough on Mark I eyeball and brain.

- AI is very useful in several situations, but appears to lack the ability to leave a troubleshooting path once started
- GPT-5.4 mini is definitely not useful for technical troubleshooting

Solution

After recovering from backup, I finally checked disk space and found that the root partition was almost full (likely completely full during the original problem). I tracked it down to MySQL tables associated with some of the WordPress sites – specifically those tied to a WordPress security plugin, iThemes Security. This is an excellent tool, but it builds temp, log, and locking tables and never cleans them out, so it grows without bound until it fills the disk (and `/var/lib/mysql` was on the root filesystem).

I logged into MySQL and truncated the three tables. iThemes Security continues to function, though truncating means it will "forget" who has attacked the site – you only lose a few recent entries, so it's a decent solution. If I had wanted to be neat and tidy, I would have deleted rows older than four weeks or so. I then set up `cleanWPSecLogs`, a script I wrote, to run once a week. It's free to download:

```
svn co http://svn.dailydata.net/svn/sysadmin_scripts/trunk/Wordpress
```

Related Issue

The next morning, I was awakened by a Help Desk technician telling me that `good.example.org` was offline. When I checked, it showed the same symptoms as `bad.example.org` had the previous day. I again tried troubleshooting – checking disk space this time – but found the same issue. The big clue that this was related was that `apachectl -S` again showed only one virtual site, when I knew it should show several.

Solution

I remembered that I had accidentally run the command `/root/.acme.sh/acme.sh –cron –home /root/.acme.sh` on `good.example.org` the previous day. I did some minor troubleshooting and was ready to revert to backup again, but decided to try one thing first – it wouldn't hurt, since I needed to recover from backup anyway. I ran:

```
ispconfig_update.sh --force
```

and told it to rebuild the site information. I don't remember the exact prompt, but it was something like *reconfigure services?* I believe I answered with the default.

The service came back up. When I ran `apachectl -S`, it showed all my virtual domains, and I was able to hit web sites – including the ISPConfig3 control panel – with no problems.

Final Note

When we did the backup recovery on `bad.example.org` on the first day, the database and filesystem ended up out of sync. The database files live on the hard disk, while the web site files are on an NFS mount. So the database was as of the backup date, but the files were current (we didn't touch the NFS mount during recovery). This points out an inconsistency in our backup system: the live site is on NFS (which is on a ZFS filesystem we snapshot daily), but the database is on the actual hard disk (a ZFS volume), which is only snapshotted weekly. So the database and filesystem can get out of sync.

We now have a new cron job that dumps the database nightly, just before the nightly NFS snapshots are taken. That script, `backupDatabasesDaily`, is in the same Subversion directory referenced earlier.