Backporting of security fixes

Contributed by Linux
Wednesday, 18 April 2007
Last Updated Thursday, 19 April 2007

The term 'backporting' describes when we take a fix for a security issue out of the most recent version of a product, and apply that fix to an older version.

Backporting is common practice amongst software vendors such as Red Hat and is essential to ensuring that we can deploy automated updates on systems. However, backporting has not been given much attention and will be a new concept to people more familiar with proprietary software.

The term 'backporting' describes when we take a fix for a security issue out of the most recent version of a product, and apply that fix to an older version.

Backporting is common practice amongst software vendors such as Red Hat and is essential to ensuring that we can deploy automated updates on systems. However, backporting has not been given much attention and will be a new concept to people more familiar with proprietary software.

For example, let's look at Apache. Red Hat shipped Apache httpd version 2.0.40 with Red Hat Linux 8.0. Shortly after the release a number of security issues were found and disclosed by the Apache Software Foundation. The Apache Software Foundation issued a new release, Apache 2.0.43, which contained fixes for these issues.

However, in addition to security fixes, a number of other changes had been made between Apache versions 2.0.40 and 2.0.43 including some features being added. Most important is the fact that the Apache server team had also make changes to the module interface.

These module interface changes mean if Red Hat was to release Apache version 2.0.43 as a security erratum replacement for our shipped version 2.0.40, then any modules that users are using with Apache would have to be recompiled. If the proprietary modules are being used, users would have to go back to the supplier of those modules to get updates.

Automated upgrade systems like Red Hat Network would not be usable if users who installed the security fixed version of Apache could suddenly find their servers no longer working.

So in cases like this, Red Hat has another option. We can identify the security fixes, isolate them from the other changes, make sure the fixes don't introduce any unwanted side effects, and apply them to our previous released version. This is backporting. On every security issue that affects software shipped by Red Hat we analyse the changes made to see if we can update to a newer version, or if we will backport a security fix. We want as many of our customers to apply security fixes as possible, so our aim is to make it quick and safe.

We can't please everyone, and backporting annoys some users who always want to be upgraded to the latest and greatest releases. However, in general, customers are more interested in stability, and the ability to minimise the changes needed for their QA and deployment.

So backporting has a number of advantages to the user, but it can create confusion. Customers need to be aware that just looking at a version number by itself doesn't help you know if you are vulnerable to a security issue. Customers also have to be careful of stories in the press that are likely to say you need to "upgrade to Apache httpd 2.0.43 to fix the issue." In addition, some security scanning tools make decisions about vulnerabilities based solely on the version number of components they find; even established tools like Nessus will give you a false positive for many vulnerabilities in Apache, for example.

Since the start of 2003 we have been very careful to explain in our security advisories how we fixed an issue--by moving to a new upstream version of the code or by applying patches to the existing versions. We've also attached CVE names to all our advisories since January 2000, which lets customers easily cross-reference a particular vulnerability and find out when and how we fixed it, independent of version numbers. In doing this, we hope to remove some of the confusion surrounding backporting, and make it easier for users to always keep up to date with the latest security fixes.