3.6. Other Roles and Variations

There can be other roles in the CVD process too, but they tend to be subordinate to the ones already described. We discuss a few of them here.

**Users**

Individual users of vulnerable products overlap with deployers as described above. In the case where the user must trigger an update or install a patch, the user is playing the role of a deployer. In other cases, the user depends on another deployer (e.g., the user’s IT support staff or an app store’s automatic update capability). In these latter cases the user does not play as active a role in the vulnerability response process.

**Integrator**

System integrators most often can be considered as playing the deployer role; however, depending on their contractual responsibilities and business relationships, they may also play roles as vendors or even coordinators in some cases.

**Cloud and Application Service Providers**

Insofar as cloud-based services are built on traditional computing platforms, cloud service providers can be considered deployers as we’ve described above. However, as cloud-based services (e.g., software, platform, and infrastructure as a service) have risen to prominence, they have also distinguished themselves from traditional software vendors in that their development, deployment, and delivery processes for security fixes tend to be much more direct.

For many cloud providers, the number of distinct instances of their software is quite limited, and control is centralized, so there are fewer independent decision makers in the path from vulnerability report to patch deployment.

Furthermore, the prevalence of DevOps practices among such providers means that the time from code commit to last vulnerable system patched can sometimes be measured in minutes. To be sure, development and delivery processes in traditional software environments have accelerated considerably as well, but the fact that cloud service providers have direct control over the vulnerable systems makes a significant difference in their ability to mitigate vulnerabilities across all their users in short order.

**Internet of Things**

Another class of vendors are the purveyors of Internet of Things (IoT) products. The physicality of IoT products and services often places them on the opposite end of the deployment spectrum from cloud-based services.

Unlike most other devices (laptops, PCs, smartphones, tablets), many of today’s IoT products are either non-updateable or require significant effort to update. Whether we’re talking about cars, televisions, medical devices, airplanes, sensors, home automation, or industrial control systems, too often today the patch deployment process involves going out and physically touching the thing that must be updated. Systems that cannot be updated become less secure over time as new vulnerabilities are found and novel attack techniques emerge. Because vulnerabilities are often discovered long after a system has been delivered, systems that lack facilities for secure updates once deployed present a long-term risk to the networks in which they reside. This design flaw is perhaps the most significant one already found in many IoT products, and if not corrected across the board, could lead to years if not decades of increasingly insecure devices acting as reservoirs of infection or as platforms for lateral movement by adversaries of all types. Patch deployment will likely improve as more connected things get over-the-air (OTA) update capabilities, but there is already a large installed base of systems lacking such features.

Furthermore, systems at the lower end of the price range might have “fire and forget” assumptions built into their pricing model, meaning that there is neither the technical means to deliver updates nor the support capability in place to even develop them in the first place. In the long run, regulatory intervention may influence IoT vendors to improve their vulnerability response capabilities, but the gap today is large and will likely be difficult to close entirely unless market incentives shift toward more holistic and improved security posture.

Another issue with IoT devices is their supply chain, whereby the vendor of the final product actually has very little to do with the hardware, firmware, or software development of the product it sells. We frequently observe pervasive use of third-party libraries in integrated products with neither recognition of nor adequate planning for how to fix or mitigate the vulnerabilities they inevitably contain. When developers embed a library into their product, that product often inherits vulnerabilities subsequently found in the incorporated code. Although the third-party library problem is equally pervasive in the traditional computing, cloud, and mobile worlds, it is even more concerning in contexts where many libraries wind up as binary blobs and are simply included in the firmware as such. Lacking the ability to analyze this black box code either in manual source code reviews or using most code analysis tools, IoT vendors may find it difficult to examine and improve the code’s security.

**Mobile Platforms and Applications**

Mobile devices present yet another class of stakeholders that has grown distinct in recent years. The device vendors themselves are most akin to IoT vendors, but app developers can be quite a diverse bunch, ranging from very large traditional software companies, to cloud service providers, to novices with a good idea and a few hours of coding. Perhaps the most significant outstanding issue is that many mobile devices have multi-stage, vertical supply chains, each step of which can stand in the way of security updates reaching their intended beneficiaries (i.e., the users) [1]. In both the mobile and IoT spaces, high-viscosity supply chains are bad for end-user security.

**Governments**
Governments are multifaceted stakeholders in regards to cybersecurity vulnerabilities and their disclosure. While they have always had a role as owners and operators of vulnerable networks and systems, issues surrounding vulnerability discovery, coordination, disclosure, and mitigation have become increasingly important to governments worldwide.

As the industries they regulate move toward increasing connectivity, agencies with oversight responsibilities will likely see an increased demand to extend their safety monitoring to include security issues (especially for security issues that directly impact safety). To that end, changes are happening rapidly on multiple fronts. For example, in the United States recent developments include the following: The FDA Medical Device Reporting process enables oversight and detection of potential device-related safety issues [2]. The National Highway Transportation and Safety Commission (NHTSA) collects reports of vehicle safety issues, which helps to drive its investigation and recall processes [3]. The FAA offers a number of safety reporting capabilities as well [4].

Beyond just documenting observed issues, some government agencies take an active learning approach when broader engineering failures occur. The aforementioned FDA and NHTSA reporting programs serve this purpose, but other programs exist as well. For example, the National Transportation Safety Board is explicitly tasked with investigating transportation accidents, and NASA collects lessons learned in a public database [5]. This kind of continuous improvement process has demonstrated its effectiveness in a variety of environments and seems to provide a good model for cybersecurity vulnerabilities in both the private and public sectors.

The United States is not alone in realizing that vulnerability discovery, disclosure, and remediation is important to national interests. These cybersecurity issues have been global for quite some time. The EU Parliament recently held hearings on modernizing export controls and the trade in zero-day vulnerabilities [6]. Meanwhile, a quick glance at the vulnerability database catalog being developed by the FIRST gives a good indication of the international interest in this problem space [7].

References