Behavior Transparency

Author: Ted Driggs, ExtraHop Networks

Applies to topic 3: Guidelines outlining security measures that shall be applied to the federal government's use of critical software

Identifying and preventing unexpected network behaviors relies on knowing which behaviors are expected. This is challenging in environments of any size and has become more challenging with the proliferation of cloud-dependent software and devices. Defenders cannot quickly determine whether a behavior is malicious or intended by the software's manufacturer, which gives attackers more time to compromise their target unopposed.

A public database of behaviors for different pieces of software and hardware will empower network detection and segmentation to shift the balance of power from attackers to defenders. Inspired by certificate transparency, we shall call this concept **Behavior Transparency**. This paper will focus on network behaviors, but Behavior Transparency can also be applied to endpoint behaviors.

Security analysts can quickly find unexplained behaviors, and avoid duplicative investigation efforts across organizations. Security products can index the database to find behaviors that identify specific software, enabling them to build or verify an environment inventory through passive observation. Products can also use the database in automatic investigation, threat intelligence, reputation analyses, and more.

Consider the SUNBURST attack: an adversary was able to introduce lookups to a malicious domain that went undetected because customers believed it to be a deliberate vendor behavior, the vendor was unaware of the calls' existence, and no single organization realized that every Solarwinds Orion install was impacted. Solarwinds had in fact published a list of their external connections, but that was not easy for analysts or security products to find and ingest, nor was it clear how someone would escalate a discrepancy between observed and documented behavior.

Under Behavior Transparency, Solarwinds would have published a bill of behaviors for Orion which enumerated the external destinations the software was intended to contact. As Orion installs contacted the domain, security products would have noticed the recurring domain access correlated with Solarwinds Orion, and would have used the Behavior Transparency database to determine that such network activity was not a listed behavior of any version of the product. At that point, they would have contacted Solarwinds to request that they update their bill of behavior, and Solarwinds would have discovered that something was seriously amiss.

Behavior Transparency is a public, incrementally-updated database of externally-observable software behaviors. The database is maintained and distributed out-of-band from the software it describes.

Being a public database blunts supply chain attacks by making it impossible for an attacker to tamper with a bill of materials for a single customer. If the SUNBURST attackers had tried to amend Orion's bill of behaviors, any Solarwinds employee could have seen that for themselves.

Trading away some degree of precision for accuracy and simplicity is pragmatic; an SBOM may contain exact patch information for every assembly in a piece of software, while the bill of behaviors says that version 8.x.x may hit any of 20 domain names. Reducing the set of behaviors from infinity to 20 is a massive win for security tools and analysts.

Embracing incompleteness and incremental expansion massively lowers the vendor barrier to entry, and allows vendors to document their most-frequently-flagged behaviors even if they don't know every cloud service their software consumes. Vendors must not need to fear adverse consequences to their software's functionality if they publish a partial bill of behaviors, so security tools need to know and respect when a software's behavior is self-declared to be incomplete.

Avoiding version-level precision means skew between the running software and the bill of behaviors is a near non-issue. This is essential to practical out-of-band distribution, especially for software that does not follow a strict single-lineage versioning scheme. If network segmentation is enforced based on Behavior Transparency data, it should accommodate version skew by taking the union of multiple versions' bills of behavior.

Maintaining the Behavior Transparency database separate from SBOMs is essential to vendor adoption; vendors are reluctant to give attackers a public list of their software components for fear of helping attackers focus their efforts, but many of these network behaviors are already publicly documented.

The federal government should require that its critical software maintain up-to-date bills of behavior in a public clearinghouse.