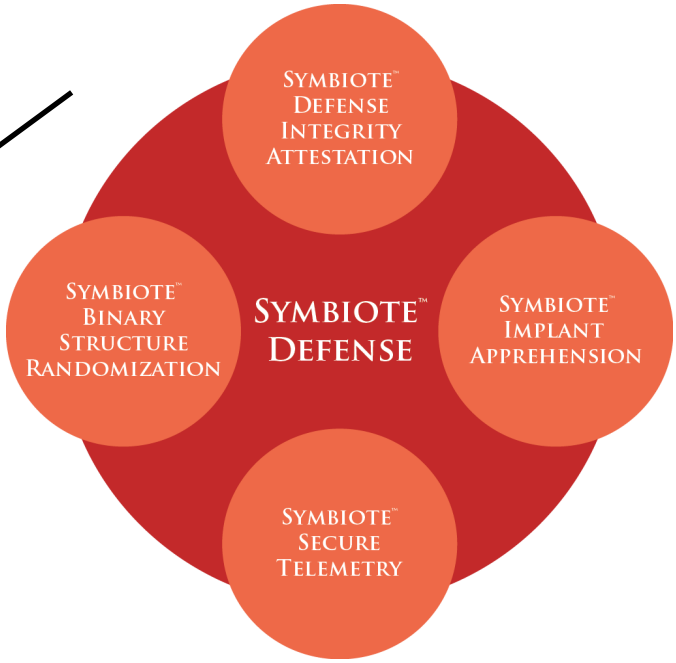
Technical Primer

How it Works

Symbiote Defense keeps embedded systems safe against attack through diversification and defensive mutualism. It defends devices without requiring any code change from the vendor or additional hardware resources, and all without impacting the functionality or speed of the device. Red Balloon Security Symbiote Defense is interlaced into the binary code and data of the host it protects. It is injected into the firmware in a randomized fashion. Once injected, it operates alongside the host program during runtime and continuously ensures that the code and data of the host device is untampered and never modified without permission. Symbiote Defense starts protecting the host the instant the host turns on, and will detect any unauthorized attempts to modify the firmware's code or data within a fraction of a second, regardless of whether the device is in sleep mode, or busy servicing requests.



Continuous Runtime Integrity Attestation

Many modern computing systems are adopting forms of load-time integrity. Load-time integrity uses a variety of protected mechanisms to prove at the time of launch that no suspicious modification has occurred. However, without Continuous Integrity Attestation, no guarantees are made on exploitation of vulnerabilities in situ. For long running systems such as ECUs and gateways, using only launch-time verification is insufficient for providing real-time situational response.

Symbiote (Continuous Runtime Attestation)	Firmware Signing
✓ Continuous real-time monitoring	• Secure Boot, Application Whitelisting
✓ Secure continuous heartbeat	• Validates Binaries at Boot or Load time
✓ Attestation uses idle cycles	• Good to have, but does not protect device at **runtime**
✓ Performance impact is customizable	• What happens if Signed Firmware has vulnerabilities?
✓ Proven Technology, 10 Billion + hours of fault-free operation	

E: info@redballoonsecurity.com T: +1 (917) 259-1057 W: www.redballoonsecurity.com

Symbiote Injection Process

Manufacturer Build Appliances are designed to operate in the existing build chain and run after compilation and prior to signing.

Manufacturer Build Appliances are appliances designed to systematically inject Symbiote Defense into firmware during the normal firmware development process. These appliances are placed after the end of the build chain and before the signing stage, and once integrated, the manufacturer's development process remains the same. A vanilla firmware bundle file comes in, and a Symbiote defended firmware bundle file goes out. All the original functionality remains the same, except now it operates more securely. Red Balloon Security does not require access to any proprietary source code, and requires no modification or addition of libraries to existing code.

These appliances are designed to scale in performance according to increasing number of cores and work better when operated in a cluster.

AESOP performs continuous state validation and forensic analysis of intrusion attempts

AESOP units perform analysis services, continuously monitoring the firmware state of Symbiote-enabled units in the field, and collecting data on intrusion attempts. The analysis is highly customizable; Red Balloon can develop machine learning algorithms to identify abnormal behavior, write code to characterize usage patterns to better understand customer needs, or even allow AESOP units to export data to Bosch's servers for more specialized analysis. AESOP can be installed on an offsite server (as shown below), stored on an embedded device on the device's network, or even installed onto the host device itself.

Build Appliance

AESOP

Products	Description
Build Appliance	Security injection appliance that instruments Symbiote Defense into firmware and randomizes layout
AESOP	Continuous firmware integrity data collector, forensic analysis and interface to security operations center

E: info@redballoonsecurity.com T: +1 (917) 259-1057 W: www.redballoonsecurity.com