DATASHEET

Serverless Protection for AWS

Protect your serverless workloads

The unstoppable migration to the cloud is seeing more businesses take advantage of serverless architectures. This adoption of Function-as-a-Service (FaaS) allows companies to deploy software faster, elastically, and at scale, without the need for any physical or virtual servers.

While many understand the technical advantages of using serverless functions and embrace the “only pay for what you use” business model of the cloud providers, there is a knowledge gap about security. More specifically, how is this code being protected from attacks?

The security challenges of serverless architecture

Many enterprises assume security is being handled by its cloud provider, but the reality is explained in the mantra that cloud providers are responsible for the ‘security of the cloud’ but not the ‘security in the cloud’. These serverless functions include code which may contain vulnerabilities and are unprotected from application attacks. These risks and attacks are included in the *OWASP Serverless Top 10*[1] and the Cloud Security Alliance’s 12 Most Critical Risks for Serverless Applications[2].

KEY CAPABILITIES:

Protects AWS Lambda functions

Supports Python, Node.js, Java and .NET

Zero day protection

Positive security model

Operates on run-time

Low operational overhead

Integrated analytics

The problems of securing serverless architectures

Decentralized controls and reduced visibility	Challenges of Traditional Security	Attack surface complexity	Maintaining secure coding controls
While serverless architecture puts developers in the front seat and allows faster deployments, security teams often don’t have monitoring abilities in these environments.	They were not built to protect ephemeral workloads on functions that are rapidly created and decommissioned, and have too many access paths.	Attackers have more to attack and exploit the architectural flexibility of the technology. For example, by poisoning the caller that triggers the serverless function to execute a malicious payload.	With the widespread use of third-party libraries, even organizations with secure coding practices struggle to manage risk found in the software supply chain.

[1] https://owasp.org/www-project-serverless-top-10/

[2] https://cloudsecurityalliance.org/blog/2019/02/11/critical-risks-serverless-applications/

Imperva Serverless Protection

Run-time security in the cloud

Imperva Serverless Protection is an innovative security solution for applications deployed in Amazon Web Services (AWS). The AWS Lambda is one of the most widely adopted technologies by enterprises and this run-time security solution prevents attacks against applications built using this serverless architecture and supports the most popular languages–Python, Node.js, Java, and .NET.

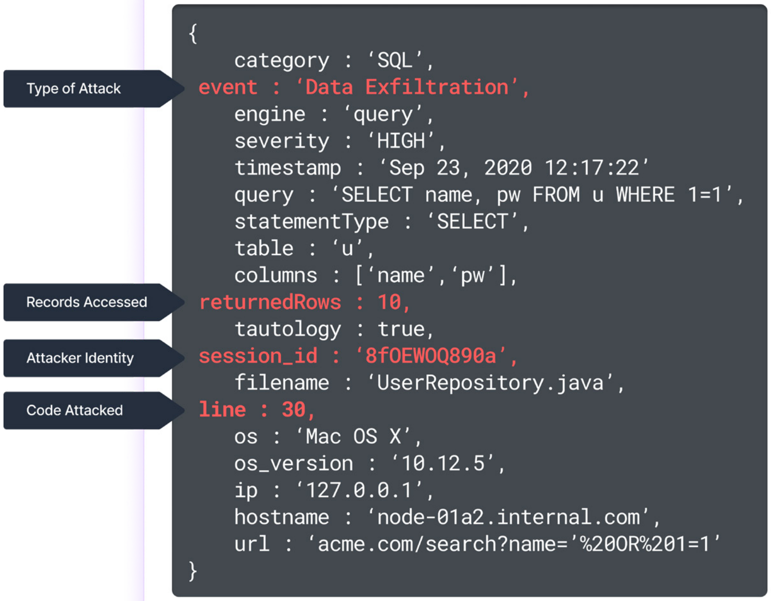
Protects against zero-day attacks

Imperva's Serverless Protection wraps around the function code and, because it operates during run-time using proprietary LangSec technologies to provide protection against zero day exploits of vulnerabilities. LangSec works by defining the scope of valid inputs and preventing applications from processing any data that falls outside of that definition.

Following a positive security model, Imperva's Serverless Protection works out of the box. Once deployed, the code is protected. There are no signature updates required or machine learning models to train. Just set it and let it protect your application.

Deep visibility into security incidents

One of the many benefits is the deep visibility into all security incidents. By monitoring during run-time we gather log-level information that allows you to fully understand the context of the attack. This information includes what caused the attack, what was accessed, and identifies the exact line of vulnerable code. Imperva's Serverless Protection also identifies and maps dependencies, such as libraries or third-party packages, used during run-time.



```
{
    category : ‘SQL’,
 event : ‘Data Exfiltration’,
    engine : ‘query’,
    severity : ‘HIGH’,
    timestamp : ‘Sep 23, 2020 12:17:22’
    query : ‘SELECT name, pw FROM u WHERE 1=1’,
    statementType : ‘SELECT’,
    table : ‘u’,
    columns : [‘name’,‘pw’],
 returnedRows : 10,
    tautology : true,
 session_id : ‘8fOEWOQ890a’,
    filename : ‘UserRepository.java’,
 line : 30,
    os : ‘Mac OS X’,
    os_version : ‘10.12.5’,
    ip : ‘127.0.0.1’,
    hostname : ‘node-01a2.internal.com’,
    url : ‘acme.com/search?name=’%20OR%201=1’
}
```

Imperva's Serverless Protection: Deep visibility into log data provided during run-time shows type of attack, records accessed, attacker identity, and code attacked.

Fully integrated with Imperva's Attack Analytics and Sonar Platform

Imperva's Serverless Protection is integrated into the Imperva Cloud Application Security solution and Attack Analytics. This provides data to correlate attacks on Imperva's Cloud Web Application and API Protection (WAAP). Imperva is unique in that it provides multi-sensor analytics powered by Attack Analytics.

Imperva Serverless Protection also works with popular technologies including Splunk, QRadar ArcSight, ELK, and AWS Security Hub.

IMPERVA CLOUD APPLICATION SECURITY

Serverless Protection is a key component of Imperva's Cloud Application Security, which reduces risk while providing an optimal user experience. Our solutions safeguard applications on-premises and in the cloud with:

Web application firewall (WAF)

Distributed Denial of Service (DDoS) protection

Advanced Bot Protection

Runtime Application Self-Protection (RASP)

Client-Side Protection

Serverless Protection

Actionable security insights

Security-enabled application delivery

Learn more about Imperva Application Security and our flexible licensing program at **imperva.com**

Imperva is an analyst-recognized, **cybersecurity leader** championing the fight to **secure data and applications** wherever they reside.