central data TECHNOLOGY
a member of CTI Group

AWS Continuum: A Smarter Approach to Application Security

Topics

1. What Is AWS Continuum?
2. Key Capabilities of AWS Continuum
3. How AWS Continuum for Code Vulnerabilities Works
4. Understanding Scan Results and Remediation
5. Supported Code Sources and Languages
6. Security, Access, and Governance
7. Implementation Considerations
8. Conclusion
9. Strengthen Your Application Security with Central Data Technology

What Is AWS Continuum?

AWS Continuum is a security service that helps organizations find and address security exposures in their applications and environments. It brings code information together with details about the infrastructure where an application runs. This context helps security and development teams understand which findings need attention first.

Modern applications can generate many security findings as teams build, update, and deploy code. However, not every finding creates the same level of risk. A weakness in an application that is exposed to a network or connected to important systems may need faster attention than an issue in a component that is not in use. AWS Continuum helps teams look beyond the number of findings and consider their possible impact.

The AWS Continuum User Guide describes three capability areas: code vulnerabilities, penetration testing, and threat modeling. This whitepaper focuses mainly on code vulnerabilities because the provided user guide explains this capability in detail. The guide refers readers to the AWS Security Agent User Guide for the detailed procedures for penetration testing and threat modeling.

central data TECHNOLOGY
a member of CTI Group

Key Capabilities of AWS Continuum

a. **Code Vulnerability Discovery and Remediation**

AWS Continuum examines first-party application code from connected repositories or uploaded code packages. It identifies potential vulnerabilities, validates and prioritizes issues, and provides remediation guidance. When a patch is available, teams can review the proposed code change and decide how to include it in their development process.

b. **Penetration Testing**

AWS Continuum also includes a penetration testing capability that continuously checks whether security weaknesses can be exploited in an environment. This helps organizations understand whether a potential issue can become a real security concern. The detailed setup and testing workflow are covered in the separate AWS Security Agent User Guide.

c. **Threat Modeling**

Threat modeling helps identify security risks in an application architecture. It considers how an application is designed and where potential threats may arise. This capability is also listed in the AWS Continuum User Guide, with its detailed instructions provided in the separate AWS Security Agent User Guide.

How AWS Continuum for Code Vulnerabilities Works

The code vulnerability workflow can be understood as a sequence of steps: connect the code, provide context, run a scan, review the findings, and work through the available patches. This gives teams a clearer process for moving from a possible vulnerability to a planned fix.

1. **Connect a code source:** Organizations can connect code stored in GitHub or GitLab, or provide code packages through Amazon S3. For GitLab, the guide supports private groups and projects, including self-managed GitLab instances that use HTTPS. The connected source gives AWS Continuum access to the code selected for scanning.

2. **Add application and infrastructure context:** AWS Continuum can use information about the application and the AWS environment where it runs. This context helps assess questions such as whether a vulnerable component is deployed, whether it may be reached over a network, and what systems could be affected if it were compromised. Infrastructure context can include AWS Config resources, AWS Lambda information, and Amazon EKS details.
3. **Run a scan:** After an environment is created, AWS Continuum starts its first scan. Teams can then choose how often later scans should run. Available options include every day, every two days, every seven days, or manual scans only. A team can also start a scan manually when needed.
4. **Review the findings:** The results are available in the AWS Continuum console. Teams can search for issues and filter them by severity to focus on the items that need attention. Each issue may include a summary, risk assessment, validation evidence, remediation information, and related findings, depending on the processing results.
5. **Review and deliver patches:** When AWS Continuum generates a patch, the team can inspect the proposed code changes and download the patch file. An optional setting can deliver patches to the connected code source. For GitHub, this creates a pull request. For Amazon S3, the patch file is uploaded to the configured bucket. Teams can then review and test the proposed change as part of their usual development process.

Understanding Scan Results and Remediation

A scan result is more useful when teams can understand why an issue matters, not only that a possible vulnerability exists. AWS Continuum provides issue details that can include severity, processing status, a short explanation of the risk, available validation evidence, and remediation information. The console supports four severity levels: Critical, High, Medium, and Low.

For example, imagine a company that maintains an online ordering application. A code scan may identify a potential weakness in one of its services. By adding infrastructure context, the team can get more information about where the service runs and whether it may be reachable through the network. This context can help the team decide whether the issue needs immediate attention and review the remediation option provided by AWS Continuum.

Infrastructure reachability information should be treated as a prioritization aid, not as a formal security guarantee. AWS documentation explains that Continuum reasons from configuration information such as security groups, routing, load balancers, and API settings; it does not perform a formal proof that a network path exists or does not exist.

Supported Code Sources and Languages

AWS Continuum supports code integrations through GitHub, GitLab, and Amazon S3. Before starting a scan, organizations should check that the selected repositories contain source files in supported formats. The guide lists C, C++, C#, Go, GraphQL, Java, JavaScript, Kotlin, PHP, Python, Ruby, Rust, Scala, Shell, SQL, Svelte, Swift, TypeScript, and Vue.

The scan uses file extensions to identify source files, and the extension must match the expected case. Some folders are skipped, including common dependency, cache, and build folders such as node_modules, .venv, __pycache__, build, dist, and target. A repository that contains only unsupported files or files inside excluded folders may not provide source files for a successful scan.

Security, Access, and Governance

Security remains a shared responsibility. AWS protects the infrastructure that runs AWS services, while customers are responsible for configuring their accounts, permissions, data access, and other security settings according to their business and regulatory requirements. Organizations should decide which users and services need access to code repositories, AWS resources, and scan results.

AWS documentation recommends protecting account credentials, using individual identities with only the permissions needed for each role, enabling multi-factor authentication, using secure connections, keeping activity logs, and applying encryption. AWS CloudTrail can be used to record API and user activity. These controls help organizations manage access and maintain visibility in their AWS environment.

When AWS Continuum collects context from an Amazon EKS cluster, the guide describes a read-only access model. The collector can read selected cluster information needed for context,

but it does not read Kubernetes Secrets and does not perform write operations. Teams should still review the required IAM roles and permissions before connecting to their environments.

Implementation Considerations

Start by identifying the applications and repositories that need to be assessed. Check supported source formats, excluded folders, required integrations, and the AWS environments that can provide useful context. This preparation helps ensure that scans have access to the code and information needed for review.

Next, set a scan schedule and decide how generated patches will be handled. Define who can review findings, approve proposed changes, and manage integrations. Include patch review, testing, and approval in the existing development process.

Conclusion

AWS Continuum helps organizations move beyond collecting vulnerability findings. For code vulnerabilities, it connects source code with application and infrastructure context, runs scans on a chosen schedule, and provides issue details and remediation patches when available. This gives security and development teams a clearer way to understand risk and plan the next steps.

Strengthen Your Application Security with Central Data Technology

Central Data Technology (CDT) can help your organization identify application security use cases, design AWS security solutions, connect code repositories and cloud environments, and plan implementation based on your requirements.

Explore how AWS security services, including AWS Continuum, may fit your environment. Contact CDT to discuss your application security needs here.

Reference: AWS Documentation