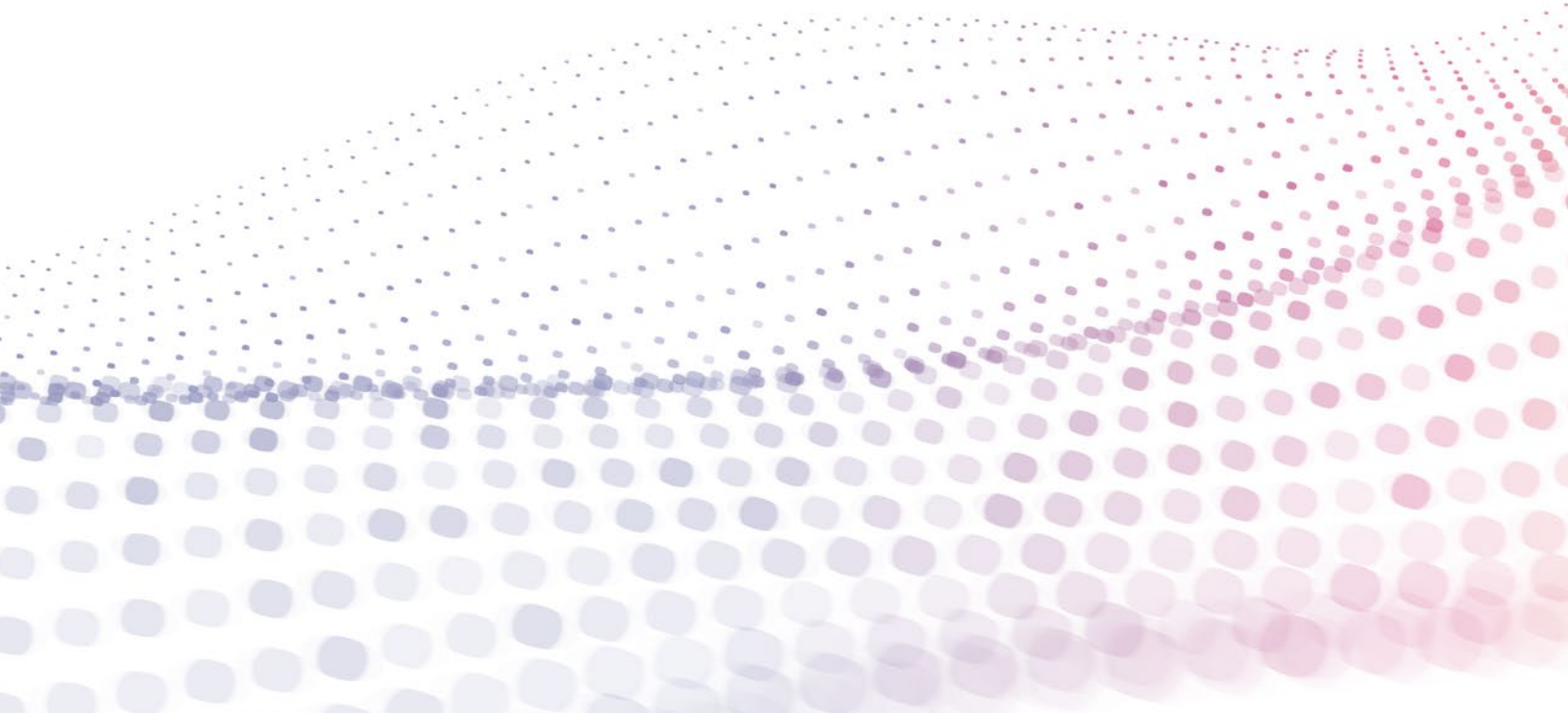
MarkLogic®

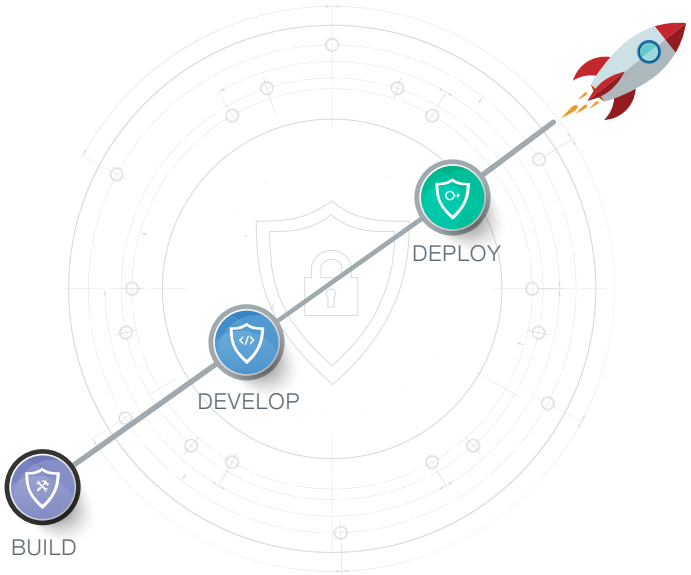
Building Security Into MarkLogic

MARKLOGIC WHITE PAPER · AUGUST 2018

With increasing attention on data security, organizations now want a better, deeper understanding of how their database secures their data at the most fundamental level. In this technical white paper, we provide an in-depth look at how we build security into MarkLogic®, including details about the *MarkLogic Security Framework* that guides the process.



Contents



Introduction

Headlines reporting cyberattacks, data held ransom, and compromises in data security are increasingly common today. As hackers and cyber terrorists become more aggressive and the risks posed by insider threats increase, organizations must take a closer look at how their data is secured.

There are many aspects to data security, but it fundamentally starts with where the data is stored, in the database. MarkLogic is a database designed for integrating, storing, managing, and searching massive amounts of enterprise data securely, so it is no surprise that we, as a company, are extremely serious about data security.

Our approach to security involves looking at an integrated security ecosystem framed by three main components:

- **How We *Build* a Secure Product**
  This area focuses on how our company's engineering team applies best practices, tools, and techniques to build the most secure product possible.
- **How to *Develop* Secure Applications on MarkLogic**
  This area focuses on the use of integrated security services and capabilities built into the MarkLogic platform that are available for use by application developers during the development lifecycle.
- **How to *Deploy* MarkLogic Securely**
  This area focuses on ensuring that MarkLogic is deployed into a secure environment. It includes the ability to work with industry-standard security technologies (e.g., LDAP, Kerberos, SSL/TLS, and KMIP) and also organizational support such as education and consulting.

In this white paper, we focus on that first aspect—how we **build** a secure product. We provide an understanding of the rigorous strategy, testing, and certifications involved in building a secure database at its most foundational level. We also provide details about the ***MarkLogic Security Framework***, a set of industry best practices, tools, and techniques used by MarkLogic's engineering team in this effort. With the assurance that comes by using a secure product, you can spend less time worrying about the security of your data, and more time getting value from it.

Key Aspects of MarkLogic's Security

Certified, Granular, Government-Grade, & Comprehensive

As a company, we focused on security from the start, building critical security features into the product starting with version 1. Without strong data security, you cannot safely store enterprise data, and that is what MarkLogic was originally designed for—storing and searching across enterprise data. There are four adjectives that best describe MarkLogic security:

- **Certified** – MarkLogic is one of only six vendors that offers a database that is Common Criteria certified, and MarkLogic is the only NoSQL database with this certification. MarkLogic also has additional certifications mentioned later in this paper.
- **Granular** – MarkLogic has Role Based Access Control (RBAC) at scale and secures data at the document and even sub-document level (similar to "cell-level" security in a relational database). With these controls, it is possible to closely govern who has access to what data across the lifecycle of data governance.
- **Government-Grade** – MarkLogic has been in the business of protecting and securing data for over a decade, and is installed and operational on classified government systems that require databases to meet extremely rigorous requirements.
- **Comprehensive** – Data governance is an end-to-end feature in MarkLogic, where data, data security, and data-driven policies are all tied together. In other words, security travels with the data.

Focused on Security From the Start

Since the first version of MarkLogic was released, we have continued to improve security in each subsequent release. See below for a sampling of key innovations added to MarkLogic, spanning over a decade.

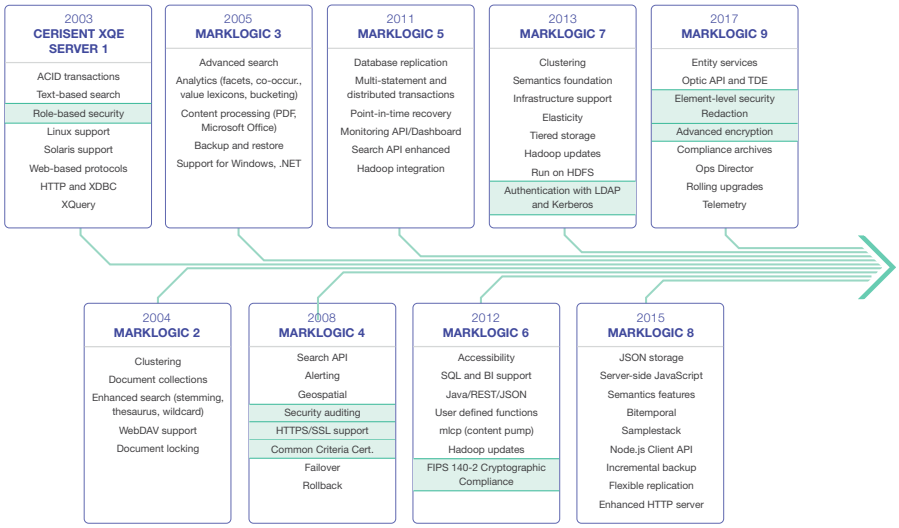


Figure 1: Continuous innovation across MarkLogic product releases

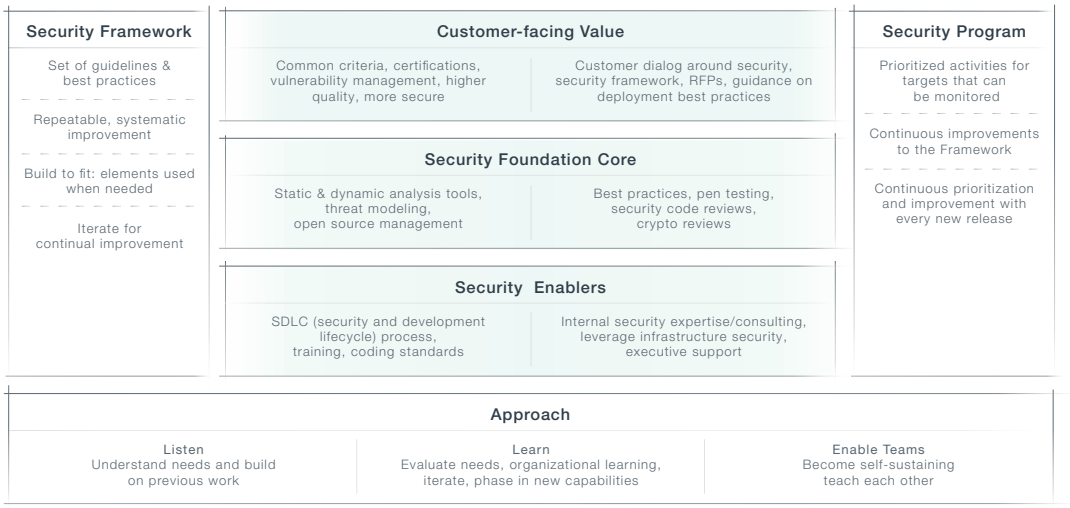


Figure 2: The MarkLogic Security Framework guides how security is built into the product

How We Build Security into MarkLogic

MarkLogic has a full host of enterprise security features that application developers and MarkLogic administrators can leverage to keep data secure. And, each of the security features in MarkLogic are built on a firm foundation: a secure MarkLogic product.

It is a basic axiom of security that security controls and security protection mechanisms must themselves be protected from attack. Otherwise, the controls and mechanisms cannot do their job and they become a hindrance. With this in mind, MarkLogic engineers make security a central concern, not a feature tacked on at the end of a release cycle.

MarkLogic Security Framework: Overview

The ***MarkLogic Security Framework*** is used by MarkLogic to build security into the product from the ground up. The Framework encompasses a set of techniques, processes, tools, automated testing and continuous integration, along with training and use of best practices.

As seen in the graphic above, the *MarkLogic Security Framework* consists of three main layers: Customer-facing Value, Security Foundation Core, and Security Enablers.

The "Customer-facing Value" layer of the *MarkLogic Security Framework* is directly visible to external stakeholders. For example, MarkLogic's Vulnerability Management process enables reporting of security issues to MarkLogic. "Security Enablers" facilitate an environment conducive to building secure products and include aspects such as training, coding standards, process, and executive focus. The "Security Foundation Core" is the engine of product security at MarkLogic.

Each component of the *MarkLogic Security Framework* enhances the effectiveness of the other components. For example, security tools and threat modeling can discover areas for follow-on code and cryptography reviews. And, training amplifies the ability of each engineer to incorporate security right form the design process.

In the following pages, we dive deeper into each part of the Framework in order to best explain the details underlying MarkLogic's comprehensive approach to ensuring a secure product.

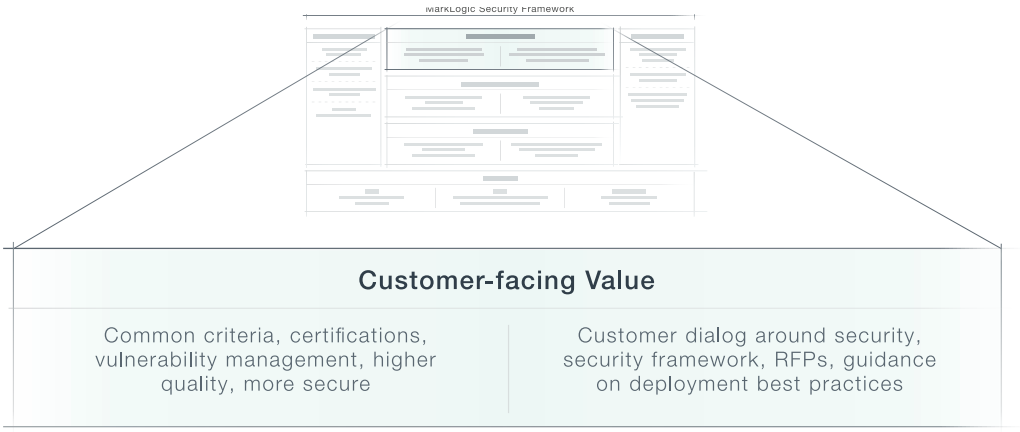


Figure 3: The MarkLogic Security Framework, with a focus on "Customer-facing Value"

MarkLogic Security Framework: Customer-Facing Value

The MarkLogic Security Framework "Customer-facing Value" components help to promote a dialog around security with our customers. Additionally, these components demonstrate externally how committed we are to building the most secure database possible.

Common Criteria Certification

The Common Criteria for Information Technology Security Evaluation (or "Common Criteria") is the driving force for the widest available mutual recognition of secure IT products worldwide. It is not easy to meet the requirements to be Common Criteria certified, and the list of vendors is short. In fact, only six DBMS vendors have products that are Common Criteria certified and MarkLogic is the only NoSQL database vendor that is part of this prestigious group.

MarkLogic was certified under one of the seventeen Certificate Authorizing Schemes and the tests were conducted by one of the largest and most widely recognized penetration testing organizations in the world. MarkLogic maintains the certification with each new release.

Common Criteria is an internationally recognized International Standards Organization standard (ISO/IEC 15408) used by governments and other organizations to assess the security capabilities of technology products. Under Common Criteria, products are evaluated according to strict standards for various features, such as security functionality and the handling of security vulnerabilities. Common Criteria gives customers more confidence in the security of technology products and helps lead to more informed decisions.

The objectives of Common Criteria are:

- To ensure that evaluations of Information Technology (IT) products and protection profiles are performed to high and consistent standards and are seen to contribute significantly to confidence in the security of those products and profiles;
- To improve the availability of evaluated, security-enhanced IT products and protection profiles;
- To eliminate the burden of duplicating evaluations of IT products and protection profiles;
- To continuously improve the efficiency and cost-effectiveness of the evaluation and certification/validation process for IT products and protection profiles.

Common Criteria certifications are recognized by 27 countries, 17 of which perform authorized certifications. (See the list of members at http://www.commoncriteriaportal.org/ccra/members/ and the Certificate Authorizing Schemes at http://www.commoncriteriaportal.org/ccra/schemes/). For a list of certified products, go to http://www.commoncriteriaportal.org/products/ and expand the section for "Databases" to see where MarkLogic is listed.

Certifications and Audits

MarkLogic is installed and operational on government systems with demanding security policies. These policies include stringent measures for access, authentication, management, audits, role separation, and system assurance. For example:

- **NIACAP** (National Information Assurance Certification and Accreditation Process) – Developed by the U.S. intelligence community for certification and accreditation of computer and telecommunications systems that handle U.S. national-security information
- **NIST Special Publication 800-37** – Guide for Applying the Risk Management Framework to Federal Information Systems; supports the six-step Risk Management Framework (RMF)

Additionally, customers have received Authority to Operate (ATO) for information systems utilizing MarkLogic that involve almost all of the major systems security standards. These standards continue to evolve and MarkLogic stays up to date on the latest changes (for example, SSAE 18 replaced SSAE 16). The system security standards currently in place on systems running MarkLogic include the following:

- NIST 800-53
- ICD 503
- FIPS 140-2
- HIPAA
- SOX 302/404
- FedRAMP
- SSAE 18
- EU 95/46/EC

Vulnerability Management

While every vendor strives to create perfect software, it is very difficult to accomplish in reality. Therefore, it is important to ensure that vendors have a mechanism in place so that security vulnerabilities can be easily reported and addressed as soon as possible.

Vulnerability Management is a process which enables input from security researchers and "white hat" security experts who want to report security issues to vendors and help them ensure that their products are highly secure.

The International Standards Organization (ISO) has published two relevant standards for Vulnerability Management:

- ISO 29147, Vulnerability Disclosure Overview, published in 2014
- ISO 30111, Vulnerability Handling Processes Overview, published in 2013

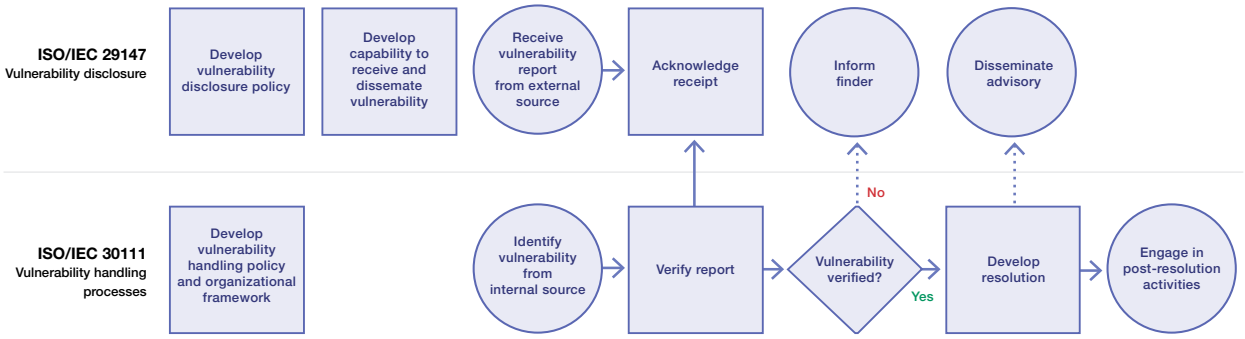


Figure 4: ISO Standards for Vulnerability Management (Source: ISO)

MarkLogic's Vulnerability Management process is based on the ISO standards, ISO 29147 and ISO 30111, which state the following:

- Vendors should have a clear way to receive vulnerability reports
- Vendors should acknowledge receipt of vulnerability reports within seven calendar days
- Vendors should coordinate with finders
- Vendors should issue advisories that contain useful information such as a unique identifier, affected products, impact and severity if the vulnerability is exploited, and how to eliminate or mitigate the issue (guidance or patching instructions)
- Generally, it is a good idea to give finders credit in the advisory if the finder wishes to be publicly acknowledged

These standards provide guidance for the process established by MarkLogic for external "finders" to identify and report potential security vulnerabilities in MarkLogic products to MarkLogic.

A "finder," according to the ISO 29147 standard, is an "individual or organization that identifies a potential vulnerability in a product or online service." Finders can be researchers, security companies, users, governments, or coordinators.

MarkLogic recognizes and appreciates the work done by finders and quickly responds to reports of security vulnerabilities in the MarkLogic product. When a finder identifies and reports a potential vulnerability, MarkLogic establishes a viable and secure channel of communication with the finder while verifying the issue being reported.

After verification, MarkLogic evaluates the potential impact to every reported issue and evaluates the potential impact to MarkLogic customers, prioritizing response based on potential customer impact. Remediation ranges from providing a hotfix or a patch to writing a knowledge base article describing workarounds and configuration changes. In all cases, MarkLogic believes security to be of the utmost importance and prioritizes remediation appropriately.

MarkLogic is a CVE Numbering Authority (CNA) and works closely with non-profit security organizations, Mitre and CERT, to address vulnerability issues promptly and effectively across the technology industry.

MarkLogic's Vulnerability Management process is available at http://www.marklogic.com/security-reporting and conforms to these guidelines.

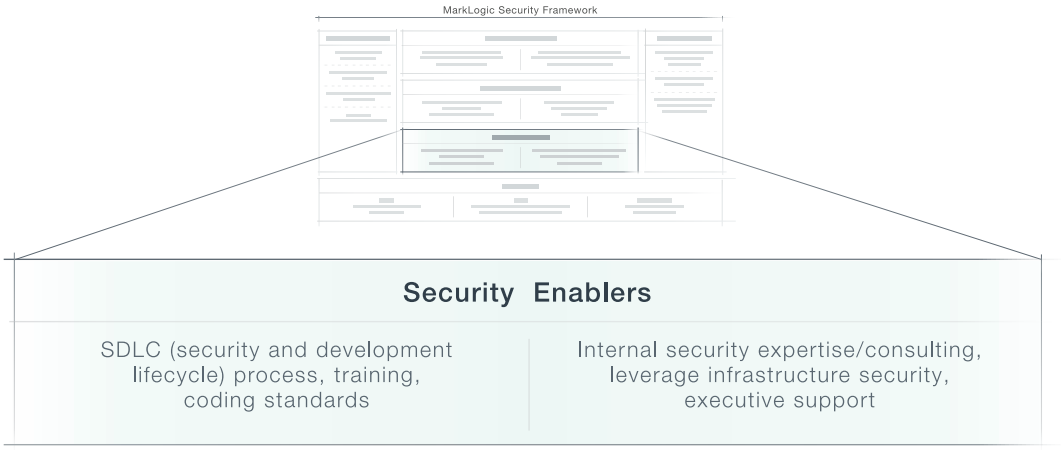


Figure 5: The MarkLogic Security Framework, with a focus on "Security Enablers"

MarkLogic Security Framework: Security Enablers

Underlying the *MarkLogic Security Framework* are the enabling elements driven by an executive vision for security. In the *MarkLogic Security Framework*, the "Security Enablers" increase the effectiveness and impact of the other components of the Framework.

Executive Focus and Vision

MarkLogic executives have extensive experience running large organizations with a focus on mission-critical infrastructure. MarkLogic leadership has consistently prioritized the investment in security to protect our commercial and public sector customers, and their mission-critical data. MarkLogic has full-time security engineers on staff, uses industry leading security tools, and uses external security experts for penetration testing and certifications.

Software Development Process

The MarkLogic Software Development Lifecycle (SDLC) supports rapid development while maintaining meaningful communication with customers through stable product roadmaps, combining agility with predictability. Frequent customer touch points have been incorporated into the process to enable MarkLogic customers to interact with new features long before release. Organizations can begin planning for deployment of a new release long before general availability (GA), lowering risk and cost for customer upgrades.

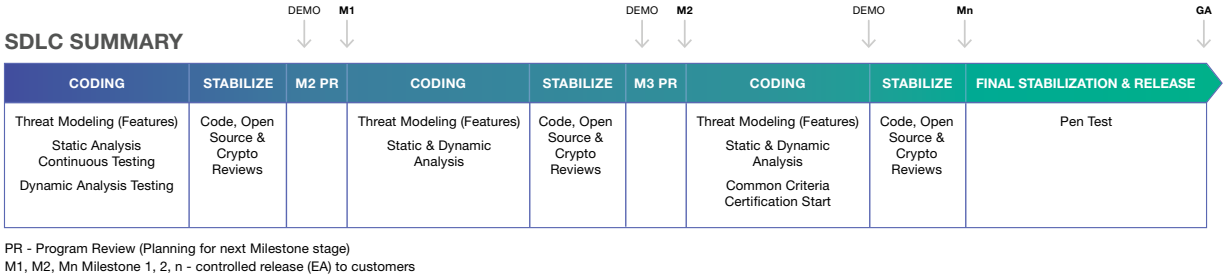


Figure 6: A point-in-time snapshot of MarkLogic Software Development Lifecycle (SDLC)

There are parallel development efforts with several major versions of MarkLogic under development (maintenance and features) at any one time.

Incorporated into the SDLC are periods dedicated to addressing technical debt, bug fixing (including any issues that are discovered as a result of the continuous integration and automated testing from static analysis tools), as well as security reviews such as threat modeling, code and cryptography reviews, etc. New features are shown to product stakeholders who provide timely input through frequent demos given by developers.

The MarkLogic SDLC promotes close collaboration among functional areas, including all relevant participants from Development, Product Management, Quality Assurance, Documentation and Support. Architecture, design, functional, and security reviews are conducted in a highly collaborative, cross-functional manner to ensure diversity of input and understanding and completeness of requirements and design.

Security Training

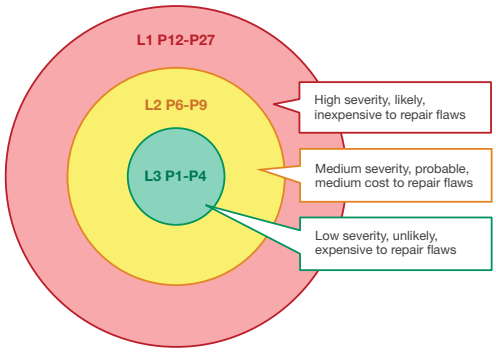
MarkLogic requires a base level of ongoing security training for the Engineering team, and makes available supplemental additional security training for all technical staff. The curriculum includes a mix of training from in-house and external industry-renowned experts. The topics covered include, but are not limited to, the following:

- General security principles
- Secure coding
- Threat modeling
- Web vulnerabilities
- Static and dynamic analysis tool usage
- Open source security
- Management and tool usage
- Recognition and mitigation of the OWASP top 10 list of vulnerabilities

MarkLogic engineers are taught how to research the latest security information available from well-known security sites such as Mitre, NVD (National Vulnerability Database), OWASP, and others, and are assisted by full-time MarkLogic security experts.

Coding Standards

MarkLogic is primarily written in C++, a language known to promote both high performance, safety, and reliability. MarkLogic uses coding paradigms recommended and advocated by architects and experts of the language (examples include Bjorne Stroustrup, Scott Myers, Herb Sutter, Robert C. Seacord, and others).



SEVERITY

VALUE	MEANING	VULNERABILITIES
1	Low	Denial of service attack, abnormal termination
2	Medium	Data integrity violation, unintentional information disclosure
3	High	Run arbitrary code

LIKELIHOOD

VALUE	MEANING
1	Unlikely
2	Probable
3	Likely

REMEDIATION COST

VALUE	MEANING	DETECTION	CORRECTION
1	High	Manual	Manual
2	Medium	Automatic	Manual
3	Low	Automatic	Automatic

Figure 7: CERT coding standards, priorities and levels

Coding standards at MarkLogic are an evolving, living set of guidelines which incorporate ideas and guidance from internal MarkLogic sources plus ideas from the CERT C++ Coding Standard. This is supplemented with elements from various other industry coding standards such as HICPP (High Integrity C++ Coding Standard V4.0), MISRA and the Lockheed JSF (Joint Strike Fighter Air Vehicle) C++ Coding Standard, among others. MarkLogic uses these coding standards to develop guidelines appropriate to developing a highly reliable and scalable database.

The CERT coding standard has been incorporated into mandatory training on C++ secure coding taken by MarkLogic engineers. It categorizes its rules and recommendations by Impact (Severity), Likelihood, and Remediation Cost so that developers can more easily prioritize which areas of programming best practices need the greatest focus.

Testing

MarkLogic uses a state-of-the-art test harness to support a continuous integration testing approach. The complete set of tests are run for all operating system platforms on a daily basis and are exercised at any time by any developer who wants to test new code before check-in.

Per industry best practices, MarkLogic uses a continuous integration approach for building, testing, and reviewing code. Developers access a test "sandbox" consisting of significant tests that are executed in a reasonably short amount of time which enables real-time integration by developers during check-in. Fully automated regression testing with over 400,000 tests is run nightly on multiple operating system platforms and in multiple configurations.

Security testing such as static analysis and open source management scanning tools are also automated and are run as a regular part of the daily cycle. Longer-running performance regression and stress tests are run continually throughout the development cycle. Tests are also available in GitHub for Java and Node.js Client APIs.

Extensive performance and stress testing is a critical part of the entire testing effort. Test plans are developed during the planning stage, and are executed continuously throughout the release.

CERT coding standards

CERT's mission is to be "a trusted provider of operationally relevant cyber security research and innovative and timely solutions to our nation's cyber security challenges. Through operationally relevant cyber security research, innovative and timely responses to cyber security challenges, and broad transition to stakeholder communities, CERT develops, executes, and evolves a technical agenda that brings unique solutions to cyber security challenges that measurably improve the security of the cyber environment." (Note: CERT is not an acronym).

CERT has worked with the Department of Homeland Security (DHS) to create US-CERT to help prevent cyber attacks, protect systems, and respond to the effects of cyber attacks across the internet.

For more information on the organization CERT, go to https://www.cert.org/. For more technical information on CERT coding standards, go to https://www.securecoding.cert.org/.

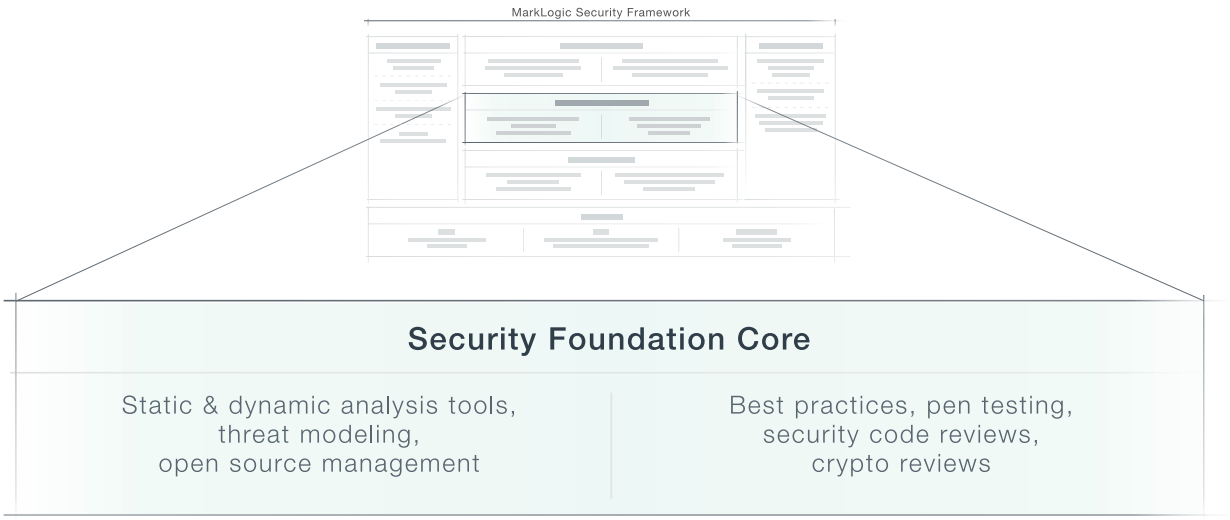


Figure 8: The MarkLogic Security Framework, with a focus on the "Security Foundation Core"

MarkLogic Security Framework: Security Foundation Core

The "Security Foundation Core" consists of industry standard practices (supported by tools and automation) that are the engine of product security. These tried and true techniques significantly increase the security of the MarkLogic product while reducing the impact and risk of any potential security issues.

Static Analysis

Code scanning soon after code check-in is one of the most efficient and thorough methods of reducing code errors. A static analysis tool can detect issues not easily found by human developers. The industry-leading tool MarkLogic uses employs *control flow analysis* to evaluate source code for potential issues such as uninitialized variables, null pointer references, web application security flaws, thread concurrency, and memory leaks across execution paths (e.g., case statement, branch, or loop), which can then be easily fixed.

MarkLogic has fully integrated static analysis into its build and test environment, performing code scans with each nightly build in order to get real-time updates on coding errors during development. Developers get immediate and personalized notifications about work-in-progress code so they can immediately fix any discovered bugs.

Open Web Application Security Project (OWASP)

OWASP is a non-profit organization dedicated to providing unbiased, practical information about application security. OWASP provides freely-available articles, methodologies, documentation, tools, and technologies in the field of web application security.

The OWASP Top Ten is particularly helpful. From OWASP: "The OWASP Top Ten is a powerful awareness document for web application security. The OWASP Top Ten represents a broad consensus about what the most critical web application security flaws are. Project members include a variety of security experts from around the world who have shared their expertise to produce this list."

For more information, visit https://www.owasp.org/index.php/Top_10.

Dynamic Analysis

Dynamic analysis is the testing and evaluation of a program in real-time with real data and is primarily used to test web-based interfaces and associated server-side application code. The objective is to identify and remediate security issues in web applications before malicious agents can exploit them.

A dynamic analysis test communicates with a web application through a front-end web proxy to identify potential security vulnerabilities and architectural weaknesses in the web application by actually performing attacks against the set of links and entry points discovered in the discovery phase of the analysis.

Dynamic analysis can:

- Test running applications and systematically find software security vulnerabilities
- Run automated scans similar to manual penetration testing performed by experts so that a comprehensive security review of all application interfaces can be obtained
- Ensure that that the entire product surface area is probed for web-based threats

MarkLogic uses several industry-leading tools to perform dynamic analysis and penetration testing. Dynamic application security testing at MarkLogic is a combination of automated and manual testing for specific well-known web vulnerabilities, exercising all the MarkLogic web interfaces and consoles.

Standard tests for dynamic analysis include the most critical web application vulnerabilities and items on the OWASP Top 10. According to the 2016 Verizon Data Breach Investigations Report, web-based application attacks are now the most frequent pattern in confirmed breaches. MarkLogic uses its full suite of Dynamic Analysis tools to detect and eliminate top web application threats.

Open Source Management

More than 6,000 new open-source vulnerabilities have been reported since 2014 to the National Vulnerability Database (NVDB), maintained by the National Institute of Standards and Technology (NIST), a globally recognized, US Federal government organization of experts in cyber security.

The widespread use of open source components provides an easy target if publicly disclosed vulnerabilities are without a traditional support model. Users of open source are not always aware of new updates and newly discovered vulnerabilities.

MarkLogic uses an industry-leading open source management tool to perform regular source code scans of the code base to build and maintain a current Bill of Materials (BOM) inventory of all 3rd party components (including open source). This helps with timely tracking of open source project security vulnerabilities (CVEs) and updates, making legal disclosures, and in ensuring that developers are supporting MarkLogic open source management policies. MarkLogic's review process for open source includes several functions across multiple teams (Legal, PM, Engineering, and Security teams) within MarkLogic to ensure that the requested usage of open source complies with MarkLogic policy.

MarkLogic's open source usage policies and governance are actively supported by management. MarkLogic open source usage policies require developers to get training on proper usage of open source software and on how to use the open source management tool used by MarkLogic.

Threat Modeling

Threat modeling is a form of risk assessment that models aspects of the attack and defense sides of a particular logical entity, such as a piece of data, an application, a host, a system, or an environment.

A threat is defined as the possibility of utilizing a software vulnerabilty to create an exploit to attack or compromise a valuable asset. A mitigation is a response to a potential threat by means of prevention, detection, reduction in impact, reduction in likelihood or by remediation after the fact.

The MarkLogic approach to threat modeling involves the following:

- Team approach – Developers, Architects, Quality Assurance, Product Management, and Support are included
- Data Flow Diagrams (DFD) – Start with the data flows and software which manages the data flows
- Leveraging existing threat models – Incremental and iterative
- Software-centric – Leverage expertise with MarkLogic
- Industry standards – CWE, CAPEC, and OWASP mechanisms for evaluating vulnerabilities
- STRIDE – An industry standard method of enumerating threats and determining mitigations:
  - **S**poofing
  - **T**ampering
  - **R**epudiation
  - **I**nformation disclosure (privacy breach or data leak)
  - **D**enial of Service (DoS)
  - **E**levation of privilege

MarkLogic performs threat modeling on every feature as the feature is being developed. The MarkLogic methodology centers on data flows, knowledge of the software being analyzed, and a cross-functional team approach in order to ensure that all threats are discovered, evaluated and addressed. Use of industry-standard resources from Mitre, OWASP, CERT, and others are incorporated into the threat modeling process as well.

Industry Standard Resources Provided by Mitre

MITRE is a not-for-profit organization that operates research and development centers sponsored by the US federal government. MITRE maintains the CVE, CWE, and CAPEC systems for vulnerability classification.

- **CVE (Common Vulnerabilities and Exposures)**
  https://cve.mitre.org/
- **CWE (Common Weakness Enumeration)**
  https://cwe.mitre.org/
- **CAPEC (Common Attack Pattern Enumeration and Classification)**
  https://capec.mitre.org/about/index.html

Penetration Testing

During every major release cycle, MarkLogic engages an external penetration testing firm to perform "pen testing" against the release. MarkLogic conducts its own version of penetration testing continuously during the development cycle for all releases. However, in order to ensure that we are finding and addressing as many potential security issues as possible, we engage a third party specialist to attempt to find vulnerabilities that may have been missed in previous testing and security evaluations. The results of the external penetration testing engagement are also used to validate and improve the *MarkLogic Security Framework*.

Penetration testing is a process and a set of security techniques where the product, server, and network are deliberately and thoroughly attacked by experts. Successful attacks are noted, weak spots are identified, and data leakages are catalogued. The intent is to exercise and test the effectiveness of existing security safeguards. The objective of penetration testing is to find out the vulnerable areas in a system and fix them before any external threat compromises and exploits these weaknesses.

Executing an attack is at the heart of any penetration test. Attacks exploit all the common types of potential vulnerabilities: buffer overflows, insufficient input validation, injection attacks, insufficient permissions, XSS, CSRF, Click-jacking, as well as the OWASP Top 10, a list of the most widely encountered web security vulnerabilities.

The key testing areas include the MarkLogic database, its APIs, security controls, and any potential impact that MarkLogic could have on the hosts, clusters, and networks where MarkLogic is installed.

There are several types of pen testing:

- **Black box** – No assistance or knowledge is provided by the vendor to the penetration testing expert
- **White box** – All information on how the product works is provided to the penetration testing consultants
- **Gray box** – Somewhere in between black and white box testing, sometimes referred to as gray box testing

At MarkLogic, we utilize gray box and in some cases white box penetration testing to enable the most effective security evaluation possible. The basic process for external penetration testing is the following:

1. Plan 2. Discovery 3. Attack 4. Report

Usually the penetration testing team will iterate over steps 2 and 3. In the "Plan" step, MarkLogic educates the team on MarkLogic and any new features. Additionally, all plans and objectives are finalized and documentation provided to the team. During the "Discovery" step, the team begins probing all web interfaces and MarkLogic APIs, looking for opportunities for attack and to confirm whether or not the MarkLogic interfaces work as documented.

Next, the "Attack" step begins and the penetration testers attack MarkLogic and its environment in order to find vulnerabilities and whether MarkLogic enables any exploitable attack vectors in the network and server environment in which MarkLogic runs. The process of discovery and attack is repeated as often as necessary in order to fully understand the likelihood and impact of any discovered weaknesses.

At the end of the cycle, a report is generated by the team that includes recommendations for improvement. MarkLogic evaluates these recommendations to best address the recommended improvements.

Penetration testing recommendations are reviewed by MarkLogic management, senior development, Quality Assurnace (QA) staff, and the security team to improve both MarkLogic and the *MarkLogic Security Framework*.

For more information from OWASP on penetration testing, download the following presentation: https://www.owasp.org/images/7/74/An_introduction_to_penetration_testing.pptx.

Code Reviews

MarkLogic has a strict policy on code that is checked into the code base. All code must be code reviewed by a senior developer on the Engineering team before check-in. To ensure that the code base never drifts from its quality goals, periodic stabilization periods, called "Controlled Check-In," require approval from the Quality Assurance team of all code changes in addition to passing developer code reviews. Additionally, developers are expected to adhere to best practices for C++ secure coding in the MarkLogic coding standard.

Cryptography Reviews

Cryptography usage is a key part of enabling and enforcing security in products and applications. Encryption is used to ensure confidentiality for data in motion (e.g., SSL/TLS) or data at rest (symmetric encryption). Cryptography is used to authenticate users and systems through asymmetric cryptography, PKI and digital signatures. Digests (hashes) are used to ensure message and data integrity.

Security specialists review cryptography usage in the MarkLogic code base to ensure usage of current versions of standard libraries (e.g. CryptSoft, OpenSSL, OpenLDAP) and appropriate cryptographic algorithms and key lengths. Industry guidelines on secure algorithms for each of the areas for cryptography are available from authoritative sources such as NIST (for more information on NIST, go to http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-131Ar1.pdf).

Cryptography reviews examine cryptography implementation in areas like key management—the generation, renewal, retirement, and transport of encryption keys. Use of standards for libraries, protocols and algorithms is critical to ensuring a secure implementation.

Conclusion

MarkLogic's approach to security relies on an integrated security ecosystem that involves securely building a secure product, ensuring customers can develop secure solutions using secure features, and ensuring customers can deploy those solutions securely. This white paper covers that first aspect—how we securely build MarkLogic.

In building the product, the MarkLogic engineering team leverages the *MarkLogic Security Framework* to ensure that best practices are followed and rigorous processes are adhered to. Here is a summary of some of the key things discussed in this paper that we do to make our product as secure as possible:

- **Secure coding** – All MarkLogic engineers are trained in secure coding practices, and MarkLogic also offers training on the latest security information and vulnerabilities.
- **Static analysis** – MarkLogic uses an industry-leading static analysis tool to test for vulnerabilities by scanning source code on each nightly build. This includes control flow analysis to test for issues such as uninitiated variables, null pointer references, and memory leaks.
- **Dynamic analysis** – MarkLogic also performs dynamic analysis to test web-based interfaces and associated server-side application code in real time, with real data. This identifies and prevents major security issues (e.g., OWASP Top 10) in web applications.
- **Open source management** – MarkLogic automates the tracking of third party and open-source components included with MarkLogic. Licenses are properly disclosed with every release, and security issues are addressed in a timely manner.
- **Threat modeling** – MarkLogic performs threat modeling on each feature as it is developed. Threat modeling models IT systems and software to understand potential threats, categorize possible impact, and mitigate vulnerabilities.
- **Penetration testing** – MarkLogic uses penetration testing to mimic real-world attacks in order to identify methods for circumventing security features. MarkLogic conducts two stages of "pen testing," each with four basic phases, to identify and address security issues.
- **Code and cryptography reviews** – Security specialists review cryptography usage in the MarkLogic code base to ensure usage of current versions of standard libraries and appropriate cryptographic algorithms and key lengths. Secure coding also ensures developers adhere to best practices.

Security is a huge topic, and there are many questions that are not addressed here. We encourage you to take a look at our other resources or contact us for more information.

Additional Resources

- **Presentation – Data Security in Practice**
  http://www.marklogic.com/resources/data-security-practice/
- **MarkLogic Concepts Guide on Security**
  https://docs.marklogic.com/guide/concepts/security
- **Understanding and Using Security Guide**
  https://docs.marklogic.com/guide/security

MARKLOGIC CORPORATION
999 Skyway Road, Suite 200 San Carlos, CA 94070
+1 650 655 2300 | +1 877 992 8885 | www.marklogic.com | sales@marklogic.com

999 Skyway Road, Suite 200 San Carlos, CA 94070
+1 650 655 2300 | +1 877 992 8885
www.marklogic.com | sales@marklogic.com