Securing Enterprise Applications

Version 1.1
Updated: November 20, 2014

Securosis, L.L.C. 515 E. Carefree Highway Suite #766 Phoenix, AZ 85085 **T** 602-412-3051
info@securosis.com www.securosis.com

Author's Note

The content in this report was *developed independently of any licensees*. It is based on material originally posted on the Securosis blog, but has been enhanced, reviewed, and professionally edited.

Special thanks to Chris Pepper for editing and content support.

This report is licensed by Onapsis:

Onapsis gives organizations the adaptive advantage to succeed in securing business-critical applications by combining technology, research and analytics. Onapsis enables every security and compliance team an adaptive approach to focus on the factors that matter most to their business–critical applications that house vital data and run business processes including SAP Business Suite, SAP HANA and SAP Mobile deployments.

Onapsis provides technology solutions including Onapsis X1, the de-facto SAP security auditing tool, and Onapsis Security Platform which delivers enterprise vulnerability, compliance, detection and response capabilities with analytics.

The Onapsis Research Labs provide subject matter expertise that combines in-depth knowledge and experience to deliver technical and business-context with sound security judgment. This enables organizations to efficiently uncover security and compliance gaps and prioritize the resolution within applications running on SAP platforms.

Onapsis delivers tangible business results including decreased business risk, highlighted compliance gaps, lower operational security costs and demonstrable value on investment.

Securing Enterprise Applications Table of Contents

Introduction

As an analyst and research firm focused on security, we talk to different groups within the Fortune 1000 every week. From our conversations it's clear most of these firms have gaps in their security program, specifically in and around the major enterprise applications which are core to their business. This is surprising as platforms like SAP and Oracle have been in place for more than a decade, you'd expect that every facet of security has some degree of coverage. And what's more, these firms are surprised to hear these gaps exist, thinking that their tools and processes provide complete coverage.

There are many reasons for these security gaps. A reliance on generic security tools focused on network or platform security and general unfamiliarity with products specifically designed to work with applications are the two major reasons. For example, companies often invest in generic assessment or configuration analysis tools, which don't actually provide an in-depth view of enterprise application configuration settings or best practices. In some cases they were told their SIEM would collect all application logs, but data collection methods don't gather the right information to evaluate user actions.

There are additional reasons as well. The enterprise application vendors *all* provide lists of security best practices, but don't list anything they do not sell, and seldom advise customers to *uninstall* unneeded components to reduce attack surface. Security teams know little about how application platforms work so they cannot independently identify which deployment models will work, and IT staff is not likely to volunteer suggestions that will require them to do more work. Finally, some security controls are avoided by large enterprises for fear they will break the application, limit usability, or degrade performance, none of which are acceptable. All of these reasons contribute to security and compliance gaps with enterprise application.

Supply chain management, customer relationship management, enterprise resource management, business analytics, and financial transaction management, are all multi-billion dollar application platforms unto themselves. Every enterprise depends upon them to orchestrate core business functions, and spends tens of millions of dollars on software and support. We are beyond explaining why enterprise applications need security to protect these investments — it is well established that insiders and persistent adversaries target these applications. Companies invest heavily in these applications, hardware to run them, and teams to keep them up and running. They perform extensive risk analysis on the costs of downtime. And in many cases their security investments are a byproduct of these risk profiles. Application security trends in the 1-2% range of total application investment, so we cannot say large enterprises don't take security seriously — they spend millions and hire dedicate staff to protect these platforms. That said, their investments are not always optimal

— enterprises may bet on solutions with limited effectiveness, without a complete understanding of the available options. It is time for a fresh look.

In this research paper, Building an Enterprise Application Security program, we will take a focused look at the major facets in an enterprise application security program, and make practical suggestions on how to improve efficiency and effectiveness of your security program. Or goal is to discuss specific security and compliance use cases for large enterprise applications, highlight gaps, and explain some application-specific tools to address these issues. This will not be an exhaustive examination of enterprise application security controls, rather a spotlight common deficiencies with the core pillars of security controls and products.

Executive Summary

Vendor recommended security controls and compliance requirements leave huge gaps in application security. It's not just that the these proposed controls provide minimal coverage, or that the majority of corporate security expenditures go to network, anti-spam and anti-virus, but many security tools used to cover these complex platforms don't actually address application specific issues. Most have no understanding of how the application platforms work, where security events should be collected, nor how to analyze application specific information. The reality is most security vendors have chosen to tackle a simpler problem, examining network and platform activity outside the application. In this research paper we will highlight the security gaps, and provide specific recommendation on how to address them.

Enterprise Application Security Use Cases

The following are the main use cases for enterprise application security, and where security and compliance teams focus their attention. We will use this outline to frame the discussion on the gaps between what most firms do today vs. what will offer more effective controls. Those of you who skim the top-line subject headings will miss the nuances of why enterprise applications provide challenges in the areas of data collection and analysis which you're generic security tools fail to address.

Compliance

Compliance with Sarbanes-Oxley (SOX) and the Payment Card Industry Data Security Standard (PCI-DSS) remain the primary drivers for security controls for enterprise applications. Most compliance requirements focus on baselining 'in-scope' applications — essentially configuration assessments — to ensure known problem areas are periodically verified as compliant. Compliance controls typically focus on issues of privileged user entitlements (what they can access), segregation of duties, prompt application of security patches, configuring the application to promote security, and consistency across application instances. These assessment scans demonstrate that each potential issue has a documented policy, that the policy is regularly tested, and that the company can produce a report history to show compliance over time. The audience for this data is typically the internal audit team, and possibly third-party auditors.

Change Management and Policy Enforcement

Beyond external compliance requirements enterprises adopt their own policies to reduce risk, improve application reliability, and reduce potential for fraud. These policies ensure that system and IT administrators perform their jobs — both to catch mistakes and to help detect administrative abuse of assigned privileges. Examples include removal of unneeded modules which contain known vulnerabilities, tracking *all* administrative changes, alerting on — and possibly blocking — use of inappropriate management tools, disabling IT administrators' access to application data, and detecting users or permissions which leaves 'backdoor' access to the system. All of which means these policies are specific to an individual organization, are more complex, and require a great deal more than application assessment to verify. Effective enforcement requires a combination of

assessment, continuous monitoring, and log file analysis. And let's not beat around the bush — these policies are established to keep administrators — of IT, databases, and applications — honest. The audience for these reports is typically internal audit, senior IT management, automated change management systems, and the security group.

Security

A debate has raged for 15 years about whether the greatest threat to IT is external attackers or malicious insiders. For enterprise applications the distinction is less than helpful — both groups pose serious threats. Further muddying the waters, external parties seek privileged access, so they may be *functioning* as privileged insiders even when that is an impersonation. Beyond attack detection, common security use cases include quarterly 'reconciliation' review, watching for *ad hoc* operations, requests for sensitive data at inappropriate times or from suspicious locations, and even general "what the heck is going on?" visibility into operations. These operations are commonly performed by users or application administrators. Of all the use cases we have listed, identifying suspicious acts in a sea of millions of normal transactions is the most difficult. More to the point, while compliance and policy enforcement are *preventive* operations, security is the domain of monitoring usage in near-real time. These features are not offered within the application or supporting database platform, but provided through external tools — often from the platform vendor.

Transaction Verification

As more enterprise applications serve external users through web interfaces, the problem of is fraud growing. Every web-facing service faces spoofing, tampering, and non-repudiation attacks, and often (and worst) SQL injection. When successful these attacks can create bogus transactions, take partial control of the supporting database, and cause errors. But unlike general security issues, these attacks are designed to create fraudulent transactions and constructed to look like legitimate traffic. How companies detect these situation varies — some firms have custom macros or procedures that look for errors after the fact, while others use third-party monitoring and threat intelligence services to detect attacks as they occur. These tools are designed to detect users who attempt to make the application behave in an unusual manner — relying on metadata, heuristics, and user/device attributes to uncover misuse by application users.

Use of Sensitive Information

Most enterprises monitor the use of sensitive information. This may be for compliance, as with payment data access or sensitive personal information, or it may be part of a general security policy. Typical policies cover IT administrators accessing data files, users issuing *ad hoc* queries, retrieval of "too much" information, or any examination of restricted data elements such as credit card numbers. All the other listed use cases are typically targeted at specific user or administrative roles, but

policies for *information usage* apply to all user groups. They are constructed to define uses cases which are not acceptable, and alert or block them. These controls may exist as part of the application logic, but are typically embedded into the database logic (such as through stored procedures), or provided by a third-party monitoring/masking tool deployed as a reverse proxy for the database.

Security Gaps

Enterprise applications typically address a specific business function: supply chain management, customer relations management, inventory management, general ledger, business performance management, and so on. They may support thousands of users, tie into many other application platforms, but these are *specialized* applications with very high complexity. It takes years of practice for application developers and IT staff to understand the nuances of these systems, the functional components that comprise an application, how they are configured, and what a transaction looks like.

Security tools also often specialize as well, focusing on a specific type of analysis — such as malware detection — and applying it in particular scenarios such as network flow data, log files, or binary files. But seldom do security tools focus their detection techniques at the application layer! They are generally designed to address threats across IT infrastructure at large in order to appeal to a wider security audience. A few move up the (OSI) stack to look at *generic* presentation or application layer threats. And fewer still actually have any knowledge of specific application functions to understand a complex platform like Oracle's Peoplesoft of SAP's ERP systems.

If you are running SAP or Oracle enterprise applications, we can be confident that you have many security tools at your disposal. Most vendors offer a combination of basic logging, identity, and encryption services to go along with published best security practices. But even vendor tools fail to address some of these deficiencies. In many cases the provided solutions were never designed for security at all, being intended to highlight errors or performance issues.

Security vendors pay lip service to understanding the application layer, but their competence typically ends at the network service port. Generic events and configuration data outside applications may be covered; internals generally are not. Let's dig into specific examples:

Understanding Application Usage

The biggest gap and most pressing need is that most monitoring systems do not understand enterprise applications. To continuously monitor enterprise applications you need to collect the appropriate data and then make sense of it. This is a huge problem because data collection points vary by application, and each platform speaks a slightly different 'language'. For example platforms like SAP speak in codes. To monitor SAP you need to understand SAP operation codes (*i.e.:* T-codes) and there are over one-hundred thousand different codes. Second you need to know where to collect these requests — application and database log files generally *do not provide* the

necessary information. As another example most Oracle applications rely heavily on stored procedures to efficiently process data *within the database*. Monitoring tools may see a procedure name and a set of variables in the user request, but unless you know what operation that procedure performs, you have no idea what is happening. Again you need to monitor the connection between the application platform and the database because audit logs do not provide a complete picture of events; then you need to figure out what the query, code, or procedure request means.

Traditional application security vendors who claim "deep packet inspection" for enterprise application security skirt understanding how the application actually works. Many use metadata (including time of day, user, application, and geolocation) collected from the network, possibly in conjunction with something like an SAP code, to evaluate user requests. They essentially monitor daily traffic to develop an understanding of 'normal', then attempt to detect fraud or inappropriate access without understanding the task being requested. This is certainly helpful for compliance and change management use cases, but not particularly effective for fraud or misuse detection. And it tends to generate false positive alerts. Products designed to monitor applications and databases actually understand their targeted application, and provide much more precise detection and enforcement. Building application specific monitoring tools is difficult and specialized work. But when you understand the application request you can focus your analysis on specific actions — order entry, for example — where insider fraud is most prevalent. This speeds up detection, lessens the burden of data collection, and makes security operations teams' job easier.

Application Composition

Throughout this research we use the term 'database' a lot. Databases provide the core storage, search, and data management features for applications. Every enterprise application relies on a database of some sort. In fact databases are complex applications themselves. To address enterprise application security and compliance you must address many issues and requirements for both the database and the application platforms.

Just as importantly, enterprise applications are never ‘off-the-rack’; they include many customizations and customer-specific code level modifications. These changes are made for lots of different reasons, but accommodations for a companies specific workflow and integration with other applications are commonly cited by customers. While the firms we’ve surveyed are comfortable with static and dynamic code scans to catch their custom code vulnerabilities, they acknowledge that ‘off-the-rack’ assessment scans and monitoring policies won’t cut it.

Deployment and Configuration

We seldom see two instances of the same application deployed the same. They are tailored to each company's needs, with configuration and user provisioning to support a specific set of requirements. This complicates configuration and vulnerability scanning considerably. What's more, application and

database assessment scans are very different from typical OS and network assessments, requiring different evaluation criteria to assess suitability. The differences lie in both how information is collected, and the depth and breadth of the rule set. All assessment products examine software revision levels, but generic assessment tools stop at list vulnerabilities and known issues, based exclusively on software versions. Understanding an application's real issues requires a deeper look. For example test and sample applications often introduce back doors into applications, which attackers then exploit. Software revision level cannot tell you what risks are posed by vulnerable modules; only a thorough analysis of a full software manifest can do that. Separation of duties between application, database, and IT administrators cannot be determined by scanning a network port or even hooking into LDAP — it requires interrogation of applications and persistent data storage. Network configuration deficiencies, weak passwords and public accounts, all easily spotted by traditional scanners — provided they have a suitable policy to check — but scanners do not discover data ownership rights, user roles, whether auditing is enabled, unsafe file access rights, or dozens of other well-known issues.

Data collection is the other major difference. Most assessment scans offer a basic network port scanner — for cases where agents are inappropriate — to interrogate the application. This provides a quick, non-invasive way to discover basic patch information. Application assessment scanners look for application specific settings, both on disk and within the database. These scans may be initiated by an agent on the application platform, or from a remote host over SSL/TLS. We call these "credentialed scans" because they require access to the file system or database, or to both. But to gather a complete picture of configuration settings, you need to collect information from the file system and database as well. This enables application assessment tools to fully address vendor best practices, industry best practices, and any *ad hoc* security or compliance rules the enterprise wants to validate. Generic assessment tools can cover about one-third of the total picture. Assessment scanners geared specifically for these critical business applications and databases get 70-100% depending on how they collect data and the policy set.

Application Patch Cycles

If you have an iPhone — or any Apple product, really — you will notice there is an update to one or more apps *every single day*. Enterprise applications are the opposite, which is unfortunate because their need is greater and the stakes are much higher. Many of you reading this know your enterprise applications run three to six months behind on security patches. If you are running big Oracle databases or SAP, odds are you are closer to 12 months behind. It's not that IT is ignoring the problem, or fails to understand that these patches address critical security issues, it's that the likelihood — and financial impact — of crashing the application is so well understood. Security patches rushed out the door have a bad habit of doing just that. It costs a lot of money to recover from such a failure, and all other IT work stops until the system is back online. The likelihood of an attacker breaching the system is not nearly so clear and any estimate of potential damage is at best a guess, so a security risk analysis cannot drive organization to patch quickly. Instead IT does what it

has always done: iteratively test the patch installer, then applications, on a series of test and pre-production systems, until they are satisfied they can safely roll the patch into production.

Because of the reticence to patch, companies look for workarounds to address the known security deficiencies. And there are in fact many potential workarounds for this problem, but the traditional approaches are all flawed. Feature removal, reduced Internet connectivity, blocking, manual process intervention, and prayer are all approaches we have heard. The good news is that some firms are speeding up the patching process by leveraging disruptive trends in IT: virtualization and the cloud. Some are using "canary testing", where the load balancer splits production traffic between patched and unpatched servers, with full switchover after the patch is vetted live. Others leverage the cloud or virtualization to spin up two sets of production servers, both patched and unpatched, and quickly rollback to unpatched systems in case of failure. These new approaches are not yet widely embraced.

Application Event and Log Collection

As mentioned earlier, database and application logs are typically not designed for security — they are primarily intended for IT personal to help understand performance issues and errors. They often omit important events including administrative activity, or provide a subset of the data such as before-and-after values for a transaction. They often lack filtering options to gather the subset of information you need — perhaps specific to a user or a transaction type — so you may be drinking from a proverbial firehose. In many cases the log file format can be set to `syslog`, so SIEM and log management systems can collect the data, but they often lack understanding of application-specific event data. But the real issue is performance — application logging typically increase platform overhead by 10-20%, and native database logging by 20-40%. This is simply a non-starter for many companies.

To effectively monitor, assess, and audit enterprise applications you will likely need to either build your own tools or leverage third-party products to supplement what you already have. Platform vendors know how to collect the correct information from their platforms, but gear their solutions to experts with their systems: system administrators, auditors, security professionals, and even IT administrators often lack the technical depth to leverage these tools. And as we mentioned earlier, the "best practices" vendors provide leave out a lot of helpful information, and do not recommend tools or services not available from the vendor.

Recommendations

Our goal for this paper was *not* to cover the breadth and depth of an entire enterprise application security program, rather the deficiencies at the core of your existing program. It's likely you can get better intelligence, with few false positives, at the same or a reduced cost. We have covered use cases and pointed out gaps; now it's time to offer recommendations for how to address the deficiencies. You will notice many of the gaps noted in the previous section are byproducts of either a) attackers exposing soft spots in security; or b) new security tools on the market; or c) innovation with the cloud, mobile, and analytics changing the boundaries of what is possible.

We divide our recommendations into two parts: Core elements of your current security program which could use improvement, and security capabilities that *should* be part of you're core capabilities, but are not.

Core Program Elements

- **Identity and Access Management:** Identity and authorization mapping form your critical first line of defense for application security. SAP, Oracle, and other enterprise application vendors offer identity tools to link to directory services, help with single sign-on, and help map authorizations — key to ensuring users only get data they legitimately need. Segregation of duties is a huge part of access control programs, and your vendor likely covers most of your needs from within the platform. But there is an over-reliance on basic services, and while many firms have stepped up to integrate multiple identity stores with federated identity, attackers have shown most enterprises need to improve in some areas. Properly mapping authorization rules into the supporting database, and real-time updates to roles and user de-provisioning top the list.

- **Passwords:** Passwords are simply not very good as a security control, and password rotation has never been proven to actually increase security; it turns out to actually be IT overhead for compliance's sake. Phishing has proven effective for landing malware on users' machines, enabling subsequent compromises, so we recommend two-factor authentication — **at least** for all administrative access. 2-factor is commonly available and can be integrated out-of-band to greatly increase the security of privileged accounts.

- **Mobile:** Protecting *your* users running *your* PCs on *your* network behind *your* firewalls is simply old news. Mobile devices are a modern — and prevalent — interface to enterprise applications. Most users don't wait for your IT department to make policy or supply devices — they go buy their own

and start using them immediately. It is important to consider mobile as an essential extension of the traditional enterprise landscape. These 'new' devices demand special consideration for how to deploy identity outside your network, how to *de*-provision users who have leave, and whether you need to quarantine data or apps on mobile devices. Cloud or 'edge' identity services, with token-based (typically SAML or OpenID) identity and mobile application management, should be part of your enterprise application security strategy.

- **Configuration and Vulnerability Management:** When we discussed why enterprise applications are different we made special mention several deficiencies in assessment products — particularly their ability to collect necessary information and lack of in-depth policies. But assessment is still one of the most powerful tools at your disposal, and generally the means for validating approximately 65% of security and compliance policies. It helps automate hundreds of repetitive, time-consuming, and highly technical system checks. We know it sounds cliche, but this really does save compliance and security teams time and money. These tools come with the most common security and compliance policies embedded to reduce custom development, and most provide a mechanism for non-technical stakeholders to obtain the technical data they need for reporting. You probably have a scanner in place already, but there is a good chance it misses a great deal of what tools designed specifically for your applications *can* acquire. We recommend making sure your product can obtain data from both inside and outside the enterprise application, along with a good selection of policies *specific to your key applications*. A handful of generic application policies are a strong indicator that you have the wrong tool. True enterprise assessment scanners, ones that can full assess key applications running on SAP and Oracle, allow you to include your policies into the scanner, and integrate with your systems and compliance management processes.

- **Data Encryption:** Most enterprise applications were designed and built with some data encryption capabilities. Either the application embeds its own encryption library and key management system, or it leverages the underlying database encryption engine to encrypt specific columns — or entire schemas — of data. Historically there have been several problems with this model. Many firms discovered that despite encrypted data, database indices and transaction logs contained and leaked unencrypted information. Additionally, encrypted data is stored in binary format, making it very difficult to calculate or report across. Finally, encryption has created performance and latency issues. The upshot is that many firms either turned encryption off entirely or removed it on temporary tables to improve performance. Fortunately there is an option which offers most of the security benefits without the downsides: transparent data encryption. It works underneath the application or database layer to encrypt data *before* it is stored on disk. It is faster that column encryption, transparent so no application layer changes are required, and avoids the risk of accidentally leaking data. Backups are still protected and you are assured that IT administrators cannot view readable data from disk. We recommend considering products from several application/database vendors and some third-party encryption vendors.

- **Firewalls and Network Security:** If you are running enterprise applications, you have firewalls and intrusion detection systems in place. And likely you also have next-generation firewalls, web

application firewalls, and/or data loss prevention systems protecting you applications. Because these investments are already paid for and in place, they tend to be the default answer to any application security question. The law of the instrument states that if all you have is a hammer, everything looks like a nail. The problem is that these platforms are not optimal for enterprise application security, but they are nonetheless considered essential because every current security job falls to them. Unfortunately they do a poor job with application security because most of them were designed to detect and address network misuse, but they do not understand how enterprise applications work. Worse, as we shift ever more toward virtualization and the cloud, physical networks go away, making them less useful in general. But the real issue is that a product which was not designed to understand the *application* cannot effectively monitor its use or function. We recommend looking at application-specific monitoring tools and application gateways — discussed in the next section — to detect and block application-specific attacks. We do not suggest throwing out your existing investments, but some problems are best addressed in a different fashion, and you need to balance network security against application security to be effective.

- **Capturing Logs and Audit Events:** You need audit logs for visibility into system usage, covering the areas simple assessments cannot, but many firms only capture the subset of events which are easy to get. Discussing logging with enterprise customers is difficult because most realize they do it poorly and don't want to be reminded of their SIEM and log management headaches. They often selected tools and deployed collectors before they fully understood what event data was really needed. As we mentioned in our discussion of why enterprise applications are different, as a rule application logs were not designed for security and compliance. So many firms leave application logging off, and instead use network logs to seed security systems. There are a couple of reasons not to go this way. First, many platform providers now understand that logs are used for security and audit teams more than for IT, and have adjusted log content and system performance to accommodate this. Second, third-party application and database monitoring systems handle complex data capture and filtering for you. Finally, some complex data types can be captured and correlated with other data to deliver on yesterday's promises. The available data is improving while the overhead (cost) of collection is shrinking. We see a convergence between the continually dropping cost of storage, improved scalability of SIEM and Log Management systems, and "big data" databases which make roll-your-own collection and analysis clusters feasible at a fraction of the cost of a 2-year-old data warehouse. Our research shows that enterprise security operations centers are collecting both new types of data, and from many more sources, in order to have sufficient information to detect attacks or perform forensic analysis. What was collected just a couple years ago is simply not adequate given current threats. You may feel you have a need to produce better log data for security or compliance, but we want to make clear most of the impediments enterprises had with log data collection are no longer an issue.

Overlooked Elements

- **Monitoring Enterprise Applications:** Continuous monitoring of enterprise application activity, with full understanding of how that application works, is the most common gap in enterprise security strategies. Application monitoring and database activity monitoring platforms, at minimum, capture and record *all application activity* (including administrator activity) in real time or near real time; across multiple platforms; and alert and/or block on policy violations. The tools remotely monitor application events, collected from a combination of sources, and collected in a central location for analysis. Think of them as an application specific SIEM and IDS combo. They are designed to understand how application platforms work down to the transaction level, with multiple methods (including heuristics, metadata, user behavior, attributes, command whitelists, and command blacklists) available to analyze events. These platforms are *focused* on the application layer, and designed to understand the specific nuances of these platforms to provide more granular and more effective security controls. This means that you can **block** activity, not just monitor. Properly configured with white/black listing, they help prevent exploitation of 0-day attacks and filter out other unwanted behavior. They work at the application layer so they are typically deployed one of three ways: as an agent on the application platform, as a reverse proxy for the application, or embedded into the application itself. Our recommendation is to use one of these platforms to monitor *business critical application* events; general-purpose network monitors do not fully understand applications, which causes more of both false positives (false alarms) and false negatives (missed attacks).

- **API Gateways:** In their rush to provide more services in order to promote customer usage and affinity, large enterprises have reimplemented traditional back-office applications to directly support customer-facing web applications. Many enterprise applications, designed and built before the Internet, have been recast as front-line customer-facing services. These platforms provide data and transactional services to support web applications, put their security is often a disaster. Command injection, SQL injection, remote vulnerability exploits, and compromise of administrative accounts are all common. In the last couple years, to support safe access to enterprise application services for remote users — particularly to support mobile applications — several firms have developed API gateways. They offer an abstraction layer which maps back-office application functions to the most common modern programming interface: RESTful APIs. The gateway builds in version control, testing facilities, support for third-party developers, detection of jailbroken devices and other signs of potential fraud, policy support for mobile app usage, integration with internal directory services for provisioning/de-provisioning, and token-based user credentials for improved identity management. If you intend to leverage enterprise applications to support end users we recommend moving past the simple firewall and web filtering security model to API gateways.

- **Penetration Testing:** Penetration testers offer an invaluable service, going outside the box to attack application security and find ways to bypass or break security controls. Most pen tests discover unknown defects in applications or deployments. This approach is so powerful because the tests find issues developers did not even know to look for. Enterprise applications and

databases incorporate a great deal of custom code, which naturally includes unique vulnerabilities. Attackers are good at finding these defects, with *very* powerful software tools to help them. A good tester finds these defects using the very same attack tools, and they find issues you don't have policies for, examining behavioral aspects of your code you were not even aware of. Of course people who know what they are doing cost money. But you should test on an ongoing basis anyway. Every time a new version of your application, a new server, or a change to your custom code occurs, you have potential for new vulnerabilities. We are big proponents of penetration testing, and strongly recommend having a service regularly check your sites. That said, there are many good third-party commercial and open source tools available if money is tight and you have expertise on staff.

Final Thoughts

These enterprise applications security recommendations address both interfaces and internal workings. We urge you to re-examine your security program in light of the areas discussed and take a fresh look at the controls you have in place. Shoring up deficiencies with preventative controls, opting for more appropriate security controls, and building in real-time monitoring and response will greatly improve security. We understand that some of these recommendation incrementally add to security or compliance budget, and we are not fans of telling people to "Do more with more." We appreciate how difficult security budget is to obtain, so we are very sensitive about making these types of recommendations. But the good news is that the majority of our recommendations are for platforms that *automate* existing work, bundling reporting and policies that would otherwise require manual work. And in several cases our recommendations only reallocate where existing budget is spent, using products more focused on core applications.

If you have any questions on this topic, or want to discuss your situation specifically, feel free to send us a note at info@securosis.com.

About the Analyst

Adrian Lane, Analyst/CTO

Adrian Lane is a Senior Security Strategist with 25 years of industry experience. He brings over a decade of C-level executive expertise to the Securosis team. Mr. Lane specializes in database architecture and data security. With extensive experience as a member of the vendor community (including positions at Ingres and Oracle), in addition to time as an IT customer in the CIO role, Adrian brings a business-oriented perspective to security implementations. Prior to joining Securosis, Adrian was CTO at database security firm IPLocks, Vice President of Engineering at Touchpoint, and CTO of the secure payment and digital rights management firm Transactor/Brodia. Adrian also blogs for Dark Reading and is a regular contributor to Information Security Magazine. Mr. Lane is a Computer Science graduate of the University of California at Berkeley with post-graduate work in operating systems at Stanford University.

Copyright

About Securosis

Securosis, LLC is an independent research and analysis firm dedicated to thought leadership, objectivity, and transparency. Our analysts have all held executive level positions and are dedicated to providing high-value, pragmatic advisory services. Our services include:

- **The Securosis Nexus**: The Securosis Nexus is an online environment to help you get your job done better and faster. It provides pragmatic research on security topics that tells you exactly what you need to know, backed with industry-leading expert advice to answer your questions. The Nexus was designed to be fast and easy to use, and to get you the information you need as quickly as possible. Access it at <https://nexus.securosis.com/>.

- **Primary research publishing**: We currently release the vast majority of our research for free through our blog, and archive it in our Research Library. Most of these research documents can be sponsored for distribution on an annual basis. All published materials and presentations meet our strict objectivity requirements and conform to our Totally Transparent Research policy.

- **Research products and strategic advisory services for end users**: Securosis will be introducing a line of research products and inquiry-based subscription services designed to assist end user organizations in accelerating project and program success. Additional advisory projects are also available, including product selection assistance, technology and architecture strategy, education, security management evaluations, and risk assessment.

- **Retainer services for vendors**: Although we will accept briefings from anyone, some vendors opt for a tighter, ongoing relationship. We offer a number of flexible retainer packages. Services available as part of a retainer package include market and product analysis and strategy, technology guidance, product evaluation, and merger and acquisition assessment. Even with paid clients, we maintain our strict objectivity and confidentiality requirements. More information on our retainer services (PDF) is available.

- **External speaking and editorial**: Securosis analysts frequently speak at industry events, give online presentations, and write and/or speak for a variety of publications and media.

- **Other expert services**: Securosis analysts are available for other services as well, including Strategic Advisory Days, Strategy Consulting engagements, and Investor Services. These tend to be customized to meet a client's particular requirements.

Our clients range from stealth startups to some of the best known technology vendors and end users. Clients include large financial institutions, institutional investors, mid-sized enterprises, and major security vendors.

Additionally, Securosis partners with security testing labs to provide unique product evaluations that combine in-depth technical analysis with high-level product, architecture, and market analysis. For more information about Securosis, visit our website: <http://securosis.com/>.