reading this):

• Injection
• Broken Authentication
• Sensitive Data Exposure
• XML External Entities (XEE)
• Broken Access Control
• Security Misconfiguration
• Cross-Site Scripting (XSS)
• Insecure Deserialization
• Using Components with Known Vulnerabilities
• Insufficient Logging & Monitoring

This list represents the most common vulnerabilities that reside in web-based software and are exploited most often. You can find out more information pertaining to these vulnerabilities at https://owasp.org/www-project-top-ten/.

Secure Coding Practices

So, we've talked about secure coding practices, but what exactly are they? Although the specific practices vary from one organization to the next, generally, they break down into two categories: standards and guidelines. Recall from Chapter 1 that standards are mandatory activities, actions, or rules, whereas guidelines are recommended actions and operational guides that provide the necessary flexibility for unforeseen circumstances. By enforcing secure coding standards and maintaining coding guidelines that reflect best practices, software development organizations dramatically reduce their source code vulnerabilities. Let's see how this works.

Coding Standards

Standards are the strongest form of secure coding practices because, to be considered a standard, a practice must meet the following requirements:

• Demonstrably reduce the risk of a particular type of vulnerability
• Be enforceable across the breadth of an organization's software development efforts
• Be verifiable in its implementation

EXAM TIP The rigorous application of secure coding standards is the best way to reduce source code vulnerabilities.

A very good reference for developing coding standards is the OWASP Top 10 list referenced in the previous section. Though it's focused on web applications, most of the vulnerabilities apply to any kind of software. Another good source of information is the

organization’s own past experience in developing code with vulnerabilities that later had
to be patched.
Once the vulnerabilities are identified, even if at a fairly high level, coding standards
can be developed to reduce the risk of building code that contains them. This is where
things get a bit sticky, because the standards vary from one programming language to
the next. If your organization develops web applications in Ruby (a common language
for web apps), the way in which you reduce the risk of, say, broken authentication will
be different than if you use PHP (another popular web app language). Still, there are
plenty of opportunities to build standards that apply to all languages when we take a
step back and consider the processes by which we develop, operationalize, and maintain
that code. We’ll cover this in more detail when we discuss security controls for software
development later in this chapter.
Finally, a standard is only good if we can verify that we are complying with it.
(Otherwise, why bother?) So, for instance, if we have a standard that reduces the risk of
injection by validating inputs and parameters, then we should have a way to verify that
none of our code fails to validate them. An excellent way to verify compliance with secure
coding standards is the practice of code reviews, as discussed in Chapter 18. Ideally,
though, we can verify at least some of our standards automatedly.
Coding standards enable secure coding by ensuring programmers always do certain
things and never do others. For example, a standard could require use of a particular library
for encryption functions because it’s been analyzed and determined to be sound and free
from vulnerabilities. Another example of a standard could be forbidding programmers
from using specific unsafe functions, such as the notorious strcpy() function in the
C programming language. This function copies a string from one memory location
to another, but doesn’t check the length of the string being copied compared to the
destination. If the string is longer than the destination, it will overwrite other areas of
memory, which can result in a buffer overflow condition.

Software-Defined Security
A promising new area of security builds on the idea of software-defined

networking
(SDN), which we covered in Chapter 13. Recall that, in SDN, the control plane (i.e., the routing and switching decisions) is separate from the data plane (i.e., the
packets and frames moving around). This allows centralized control of the network,
which in turn improves performance, flexibility, and security. SDN also enables the
separation of security functions from more traditional network appliance approaches.
Software-defined security (SDS or SDSec) is a security model in which security functions such as firewalling, intrusion detection and prevention (IDS/IPS), and network
segmentation are implemented in software within an SDN environment. One of the advantages of this approach is that sensors (for functions like IDS/IPS) can be
dynamically repositioned depending on the threat environment.
SDS is a new technology but promises significant security advantages. Because of its dependence on SDN, SDS is best used in cloud and virtualized network environments.

NOTE Coding standards are required in certain regulated sectors such as automobile and railroad control software, among others.

Coding Guidelines
Secure coding guidelines are recommended practices that tend to be less specific than
standards. For example, coding guidelines might encourage programmers to use variable
names that are self-explanatory and not reused anywhere else in the program because this
makes the code easier to understand. Applied to secure coding, these standards can help
by ensuring code is consistently formatted and commented, which makes the code easier
to read during code reviews. Guidelines may also recommend that coders keep functions
short (without specifying how short) because this reduces the chance of errors. These
practices may not sound like much, but they make it easier to spot errors early in the
development process, thus improving quality, while decreasing vulnerabilities and costs.

Security Controls for Software Development
We tend to think of security controls as something to be added to an environment in
order to reduce risks to it. While this is certainly true of software development environments, secure coding adds another layer, which consists of the security controls we
build into the code itself. Regardless of whether we are protecting the development
subnetwork or the software that is produced therein, we should implement security

controls only after conducting deliberate threat modeling tied to a risk analysis process.

Keep in mind, however, that the threat models for an internal subnet are different from the threat models for software you're deploying throughout your organization or even selling to your customers. Either way, the goals are to reduce vulnerabilities and the possibility of system compromise, but the manner in which we do so will be very different.

Let's zoom in on just software you're developing. Which specific software controls you should use depends on the software itself, its objectives, the security goals of its associated security policy, the type of data it will process, the functionality it is to carry out, and the environment in which it will be placed. If an application is purely proprietary and will run only in closed, trusted environments, it may need fewer security controls than those required for applications that will connect businesses over the Internet and provide financial transactions. The trick is to understand the security needs of a piece of software, implement the right controls and mechanisms, thoroughly test the mechanisms and how they integrate into the application, follow structured development methodologies, and provide secure and reliable distribution methods.

In the sections that follow, we'll identify and describe the application of security controls for the major aspects of software development. These include aspects of the software itself, of course, but also the tools used to develop it, the manner in which we test it, and even how to integrate the software development environment into the broader security architecture.

Development Platforms

PART VIII

Software is normally developed by a team of software engineers who may or may not use the same tools. The most important tool in their tool set is an integrated development environment (IDE), which enables each engineer to pull code from a repository (more

on that later), edit it, test it, and then push it into the repository so the rest of the team can build on it. Depending on the programming language, target environments, and a host of other considerations, your developers may use Eclipse, Microsoft Visual Studio, Xcode, or various other applications. The software they develop will likely be tested (formally or otherwise) using development clients and servers that are supposed to represent the production platforms on which the finished software product will run. When we talk about security of the development platforms, therefore, we mean both the development endpoints and the "fake" clients and servers on which the software gets tested.

It may seem obvious, but the first step in ensuring the security of development platforms is to secure the devices on which our software engineers practice their craft. The challenge that many organizations face is that their engineers tend to be more sophisticated than the average user and will make changes to their computers that may or may not be authorized. Their principal incentive, after all, is to develop code quickly and correctly. If the configuration of their workstation gets in the way, it may find itself being modified. To avoid this, you should resist the temptation of giving your software engineers unfettered privileged access to their own devices. Enforcing good change management practices is critical to securing these development endpoints.

Even harder than ensuring change controls on your developers' workstations is securely provisioning the development clients and servers that they will need for testing.

Many organizations allow their developers to stand up and maintain their own development environment, which may be fine provided that these devices are isolated from the production environments. It may sound like common sense, but the problem is that some organizations don't do a good enough job of isolating development and production systems. In principle, doing so simply requires putting the development nodes in an isolated VLAN. In practice, the demarcation is not that cut and dry. This gets even more challenging when the team is distributed, which requires your developers (or perhaps their external collaborators) to remotely access the development hosts.

The best solution is to require use of a VPN to connect to the isolated development

network. This may create a bit of work for the operations staff but is the only way to ensure that development and production code remains separate. Another good approach is to create firewall rules that prevent any unauthorized external connections (and even then only the bare minimum) to or from development servers. It should be clear by now that the provisioning of hosts on the development network should not be left to the software development team.

Tool Sets

As the old saying goes, you can't make everyone happy. Your IDE may be awesome, but invariably your software developers will need (or just want) additional tool sets. This is particularly true for developers that have a favorite tool that they've grown used to over the years, or if there is new work to be done for which the existing tools are not ideal.

There are two approaches we've seen adopted by many organizations, and neither is ultimately good. The first is to force strict compliance with the approved tool sets that the organization provides. On the surface, this makes sense from a security and operations perspective. Having fewer tools means more standardization, allows for more thorough security assessments, and streamlines provisioning. However, it can also lead to a loss in productivity and typically leads the best coders to give up and move on to another organization where they're allowed more freedom.

The other (not good) approach is to let the developers run amuck in their own playground. The thinking goes something like this: we let them use whatever tools they feel are good, we set up and maintain whatever infrastructure they need, and we just fence the whole thing off from the outside so nothing bad can get in. The end of that sentence should make you shake your head in disagreement because keeping all the bad stuff out obviously is not possible, as you've learned throughout this book. Still, this is the approach of many small and mid-sized development shops.

A better approach is to treat the software development department the same way we treat any other. If they need a new tool, they simply put in a request that goes through the change management process, discussed in Chapter 20. The change advisory board (CAB) validates the requirement, assesses the risk, reviews the implementation plan, and so on. Assuming everything checks out and the CAB approves, the IT operations team

integrates the tool into the inventory, updating and provisioning processes; the security
team implements and monitors the appropriate controls, and the developers get the new
tool they need.

Application Security Testing

Despite our best efforts, we (and all our programmers) are human and will make mistakes. Some of those mistakes will end up being source code vulnerabilities. Wouldn’t
it be nice to find them before our adversaries do? That’s the role of application security
testing, which comes in three flavors that you should know for the CISSP exam: static
analysis, dynamic analysis, and fuzzing.

Static Application Security Testing

Static application security testing (SAST), also called static analysis, is a technique meant to
help identify software defects or security policy violations and is carried out by examining
the code without executing the program, and therefore is carried out before the program
is compiled. The term SAST is generally reserved for automated tools that assist analysts
and developers, whereas manual inspection by humans is generally referred to as code
review (covered in Chapter 18).

SAST allows developers to quickly scavenge their source code for programming flaws
and vulnerabilities. Additionally, this testing provides a scalable method of security
code review and ensures that developers are following secure coding policies. There
are numerous manifestations of SAST tools, ranging from tools that simply consider
the behavior of single statements to tools that analyze the entire source code at once.

However, you must remember that static code analysis can never reveal logical errors
and design flaws, and therefore must be used in conjunction with manual code review to
ensure thorough evaluation.

Dynamic Application Security Testing

Dynamic application security testing (DAST), also known as dynamic analysis, refers to the
evaluation of a program in real time, while it is running. DAST is commonly carried
out once a program has cleared the SAST stage and basic programming flaws have

been
rectified offline. DAST enables developers to trace subtle logical errors in the software
that are likely to cause security mayhem later on. The primary advantage of this technique is that it eliminates the need to create artificial error-inducing scenarios. Dynamic
analysis is also effective for compatibility testing, detecting memory leakages, identifying dependencies, and analyzing software without having to access the software's actual
source code.
EXAM TIP Remember that SAST requires access to the source code, which is not executed during the tests, while DAST requires that you actually run the code but does not require access to the source code.

Fuzzing

PART VIII

Fuzzing is a technique used to discover flaws and vulnerabilities in software by sending large amounts of malformed, unexpected, or random data to the target program in
order to trigger failures. Attackers can then manipulate these errors and flaws to inject
their own code into the system and compromise its security and stability. Fuzzing tools,
aka fuzzers, use complex inputs to attempt to impair program execution. Fuzzing tools

Manual Penetration Testing
Application security testing tools, together with good old-fashioned code reviews,
are very good at unearthing most of the vulnerabilities that would otherwise go unnoticed by the software development team. As good as these tools are, however, they lack the creativity and resourcefulness of a determined threat actor. For this reason, many organizations also rely on manual penetration testing (MPT) as the final check before code is released into production environments. In this approach, an experienced red team examines the software system in its intended environment and looks for ways to compromise it. It is very common for this testing
to uncover additional vulnerabilities that cannot be detected by automated tools.

are commonly successful at identifying buffer overflows, DoS vulnerabilities, injection
weaknesses, validation flaws, and other activities that can cause software to freeze, crash,
or throw unexpected errors.

Continuous Integration and Delivery
With the advent of Agile methodologies, discussed in Chapter 24, it has become possible

to dramatically accelerate the time it takes to develop and release code. This has been
taken to an extreme by many of the best software development organizations through
processes of continuous integration and continuous delivery.
Continuous integration (CI) means that all new code is integrated into the rest of
the system as soon as the developer writes it. For example, suppose Diana is a software
engineer working on the user interface of a network detection and response (NDR)
system. In traditional development approaches, she would spend a couple of weeks
working on UI features, pretty much in isolation from the rest of the development team.
There would then be a period of integration in which her code (and that of everyone
else who's ready to deliver) gets integrated and tested. Then, Diana (and everyone else)
goes back to working alone on her next set of features. The problem with this approach
is that Diana gets to find out whether her code integrates properly only every two weeks.
Wouldn't it be nice if she could find out instantly (or at least daily) whether any of her
work has integration issues?
With continuous integration, Diana works on her code for a few hours and then
merges it into a shared repository. This merge triggers a batch of unit tests. If her code
fails those tests, the merge is rejected. Otherwise, her code is merged with everyone else's
in the repository and a new version of the entire software system is built. If there are any
errors in the build, she knows her code was the cause, and she can get to work fixing them
right away. If the build goes well, it is immediately subjected to automated integration
tests. If anything goes wrong, Diana knows she has to immediately get back to work
fixing her code because she "broke the build," meaning nobody else can commit code
until she fixes it or reverses her code merge.

Continuous integration dramatically improves software development efficiency
by identifying errors early and often. CI also allows the practice of continuous delivery
(CD), which is incrementally building a software product that can be released at any
time. Because all processes and tests are automated, you could choose to release code to
production daily or even hourly. Most organizations that practice CI/CD, however, don't
release code that frequently. But they could if they wanted to.

CI/CD sounds wonderful, so what are the security risks we need to mitigate? Because CI/CD relies heavily on automation, most organizations that practice it use commercial or open-source testing platforms. One of those platforms is Codecov, which was compromised in early 2021, allowing the threat actor to modify its bash uploader script. This is the script that would take Diana's code in our earlier example and upload it for testing and integration. As an aside, because the tests are automated and don't involve actual users, developers typically have to provide access credentials, tokens, or keys to enable testing. The threat actor behind the Codecov breach modified the bash uploader so that it would exfiltrate this access data, potentially providing covert access to any of the millions of products worldwide that use Codecov for CI/CD.

The Codecov breach was detected about three months later by an alert customer who noticed unusual behavior in the uploader, investigated it, and alerted the vendor to the problem. Would you be able to tell that one of the components in your CI/CD toolset was leaking sensitive data? You could if you practice the secure design principles we've been highlighting throughout the book, especially threat modeling, least privilege, defense in depth, and zero trust.

Security Orchestration, Automation, and Response

PART VIII

The Codecov breach mentioned in the previous section also highlights the role that a security orchestration, automation, and response (SOAR) platform can play in securing your software development practices. Chapter 21 introduced SOAR in the context of the role of a security information and event management (SIEM) platform in your security operations. Both SOAR and SIEM platforms can help detect and, in the case of SOAR, respond to threats against your software development efforts. If you have sensors in your development subnet (you did segment your network, right?) and a well-tuned SOAR platform, you can detect new traffic flowing from that subnet (which shouldn't be talking much to the outside world) to a new external endpoint. If the traffic is unencrypted (or you use a TLS decryption proxy to do deep packet inspection), you'd notice access tokens and keys flowing out to a new destination. Based on this observation, you could declare

an incident and activate the playbook for data breaches in your SOAR platform. Just like
that, you would've stopped the bleeding, buying you time to figure out what went wrong
and how to fix it for the long term.
One of the challenges with the scenario just described is that many security teams
treat their organization's development environment as a bit of a necessary chaos that
must be tolerated. Software developers are typically rewarded (or punished) according
to their ability to produce quality code quickly. They can be resistant (or even rebel
against) anything that gets in the way of their efficiency, and, as we well know, security
tends to do just that. This is where DevSecOps (discussed in Chapter 24) can help build

the right culture and balance the needs of all teammates. It can also help the security
team identify and implement controls that mitigate risks such as data breaches, while
minimally affecting productivity. One such control is the placement of sensors such as
IDS/IPS, NDR, and data loss prevention (DLP) within the development subnets. These
systems, in turn, would report to the SOAR platform, which could detect and contain
active threats against the organization.

Software Configuration Management

Not every threat, of course, is external. There are plenty of things our own teammates
can do deliberately or otherwise that cause problems for the organization. As we'll see
later in this chapter when we discuss cloud services, improper configurations consistently
rank among the worst threats to many organizations. This threat, however, is a solved
problem in organizations that practice proper configuration management, as we covered
in Chapter 20.
Anticipating the inevitable changes that will take place to a software product during its
development life cycle, a configuration management system should be put into place that
allows for change control processes to take place through automation. Since deploying
an insecure configuration to an otherwise secure software product makes the whole thing
insecure, these settings are a critical component of securing the software

development
environment. A product that provides software configuration management (SCM) identifies
the attributes of software at various points in time and performs a methodical control
of changes for the purpose of maintaining software integrity and traceability throughout
the software development life cycle. It tracks changes to configurations and provides the
ability to verify that the final delivered software has all of the approved changes that are
supposed to be included in the release.
During a software development project, the centralized code repositories are often kept
in systems that can carry out SCM functionality. These SCM systems manage and track
revisions made by multiple people against a single master set and provide concurrency
management, versioning, and synchronization. Concurrency management deals with the
issues that arise when multiple people extract the same file from a central repository and
make their own individual changes. If they were permitted to submit their updated files
in an uncontrolled manner, the files would just write over each other and changes would
be lost. Many SCM systems use algorithms to version, fork, and merge the changes as
files are checked back into the repository.
Versioning deals with keeping track of file revisions, which makes it possible to “roll
back” to a previous version of the file. An archive copy of every file can be made when it
is checked into the repository, or every change made to a file can be saved to a transaction
log. Versioning systems should also create log reports of who made changes, when they
were made, and what the changes were.
Some SCM systems allow individuals to check out complete or partial copies of the
repositories and work on the files as needed. They can then commit their changes back to
the master repository as needed and update their own personal copies to stay up to date
with changes other people have made. This process is called synchronization.

Code Repositories
A code repository, which is typically a version control system, is the vault containing the
crown jewels of any organization involved in software development. If we put on

our
adversarial hats for a few minutes, we could come up with all kinds of nefarious scenarios
involving these repositories. Perhaps the simplest is that someone could steal our source
code, which embodies not only many staff hours of work but, more significantly, our
intellectual property. An adversary could also use our source code to look for
vulnerabilities to exploit later, once the code is in production. Finally, adversaries could deliberately
insert vulnerabilities into our software, perhaps after it has undergone all testing and is
trusted, so that they can exploit it later at a time of their choosing. Clearly, securing our
source code repositories is critical.
Perhaps the most secure way of managing security for your code repositories is to implement them on an isolated (or "air-gapped") network that includes the development, test, and QA environments. The development team would have to be on this network to do their work, and the code, once verified, could be exported to
the production servers using removable storage media. We already presented this best

Software Escrow

PART VIII

If a company pays another company to develop software for it, it should have some
type of software escrow in place for protection. We covered this topic in Chapter
23 from a business continuity perspective, but since it directly deals with software
development, we will mention it here also.
In a software escrow framework, a third party keeps a copy of the source code, and possibly other materials, which it will release to the customer only if specific
circumstances arise, mainly if the vendor who developed the code goes out of business or for some reason is not meeting its obligations and responsibilities. This
procedure protects the customer, because the customer pays the vendor to develop software code for it, and if the vendor goes out of business, the customer otherwise
would no longer have access to the actual code. This means the customer code could
never be updated or maintained properly.
A logical question would be, "Why doesn't the vendor just hand over the source code to the customer, since the customer paid for it to be developed in the first
place?" It does not always work that way. The code may be the vendor's intellectual
property. The vendor employs and pays people with the necessary skills to develop
that code, and if the vendor were to just hand it over to the customer, it could

be giving away its intellectual property, its secrets. The customer oftentimes gets
compiled code instead of source code. Compiled code is code that has been put
through a compiler and is unreadable to humans. Most software profits are based
on licensing, which outlines what customers can do with the compiled code. For an
added fee, of course, most custom software developers will also provide the source,
which could be useful in sensitive applications.

practice in the preceding section. The challenge with this approach is that it severely
limits the manner in which the development team can connect to the code. It also
makes it difficult to collaborate with external parties and for developers to work from
remote or mobile locations.
A pretty good alternative would be to host the repository on the intranet, which
would require developers to either be on the local network or connect to it using a VPN
connection. As an added layer of security, the repositories can be configured to require
the use of Secure Shell (SSH), which would ensure all traffic is encrypted, even inside
the intranet, to mitigate the risk of sniffing. Finally, SSH can be configured to use public
key infrastructure (PKI), which allows us to implement not only confidentiality and
integrity but also nonrepudiation. If you have to allow remote access to your repository,
this would be a good way to go about it.
Finally, if you are operating on a limited budget or have limited security expertise in
this area, you can choose one of the many web-based repository service providers and
let them take care of the security for you. While this may mitigate the basic risks for
small organizations, it is probably not an acceptable course of action for projects with
significant investments of intellectual property.

Software Security Assessments

We already discussed the various types of security assessments in Chapter 18, but let’s
circle back here and see how these apply specifically to software security.
Recall from previous sections in this chapter that secure software development practices originate in an
organizational policy that is grounded in risk management. That policy is implemented
through secure coding standards, guidelines, and procedures that should result in secure
software products. We verify this is so through the various testing methods

discussed in this chapter (e.g., SAST and DAST) and Chapter 24 (e.g., unit, integration, etc.). The purpose of a software security assessment, then, is to verify that this entire chain, from policy to product, is working as it should.

When conducting an assessment, it is imperative that the team review all applicable documents and develop a plan for how to verify each requirement from the applicable policies and standards. Two areas that merit additional attention are the manner in which the organization manages risks associated with software development and how it audits and logs software changes.

Risk Analysis and Mitigation

Risk management is at the heart of secure software development, particularly the mapping between risks we've identified and the controls we implement to mitigate them.

This is probably one of the trickiest challenges in secure software development in general, and in auditing it in particular. When organizations do map risks to controls in software development, they tend to do so in a generic way. For example, the OWASP Top 10 list is a great starting point for analyzing and mitigating vulnerabilities, but how are we doing against specific (and potentially unique) threats faced by our organization?

Threat modeling is an important activity for any development team, and particularly in DevSecOps. Sadly, however, most organizations don't conduct threat modeling for

their software development projects. If they're defending against generic threats, that's good, but sooner or later we all face unique threats that, if we haven't analyzed and mitigated them, have a high probability of ruining our weekend.

Another area of interest for assessors are the linkages between the software development and risk management programs. If software projects are not tracked in the organization's risk matrix, then the development team will probably be working in isolation, disconnected from the broader risk management efforts.

Change Management

Another area in which integration with broader organizational efforts is critical to secure software development is change management. Changes to a software project that may

appear inconsequential when considered in isolation could actually pose threats when analyzed within the broader context of the organization. If software development is not integrated into the organization's change management program, auditing changes to software products may be difficult, even if the changes are being logged by the development team. Be that as it may, software changes should not be siloed from overall organizational change management because doing so will likely lead to interoperability or (worse yet) security problems.

Assessing the Security of Acquired Software

PART VIII

Most organizations do not have the in-house capability to develop their own software systems. Their only feasible options are either to acquire standard software or to have a vendor build or customize a software system to their particular environment. In either case, software from an external source will be allowed to execute in a trusted environment. Depending on how trustworthy the source and the code are, this could have some profound implications to the security posture of the organization's systems. As always, we need to ground our response on our risk management process.

In terms of managing the risk associated with acquired software, the essential question to ask is, "How is the organization affected if this software behaves improperly?" Improper behavior could be the consequence of either defects or misconfiguration. The defects can manifest themselves as computing errors (e.g., wrong results) or vulnerability to intentional attack. A related question is, "What is it that we are protecting and this software could compromise?" Is it personally identifiable information (PII), intellectual property, or national security information? The answers to these and other questions will dictate the required thoroughness of our approach.

In many cases, our approach to mitigating the risks of acquired software will begin with an assessment of the software developer. Characteristics that correlate to a lower software risk include the good reputation of the developer and the regularity of its patch pushes. Conversely, developers may be riskier if they have a bad reputation, are small or new organizations, if they have immature or undocumented development processes, or if

their products have broad marketplace presence (meaning they are more lucrative targets
to exploit developers).
A key element in assessing the security of acquired software is, rather obviously, its
performance in an internal assessment. Ideally, we are able to obtain the source code

from the vendor so that we can do our own code reviews, vulnerability assessments, and
penetration tests. In many cases, however, this will not be possible. Our only possible
assessment may be a penetration test. The catch is that we may not have the in-house
capability to perform such a test. In such cases, and depending on the potential risk
posed by this software, we may be well advised to hire an external party to perform
an independent penetration test for us. This is likely a costly affair that would only be
justifiable in cases where a successful attack against the software system would likely lead
to significant losses for the organization.
Even in the most constrained case, we are still able to mitigate the risk of acquisition.
If we don’t have the means to do code reviews, vulnerability assessments, or penetration
tests, we can still mitigate the risk by deploying the software only in specific subnetworks,
with hardened configurations, and with restrictive IDS/IPS rules monitoring its behavior.
Though this approach may initially lead to constrained functionality and excessive false
positives generated by our IDS/IPS, we can always gradually loosen the controls as we
gain assurances that the software is trustworthy.

Commercial Software
It is exceptionally rare for an organization to gain access to the source code of a commercial-off-the-shelf (COTS) product to conduct a security assessment of it. However,
depending on the product, we may not have to. The most widely used commercial
software products have been around for years and have had their share of security researchers
(both benign and malicious) poking at them the whole time. We can simply research
what vulnerabilities and exploits have been discovered by others and decide for ourselves
whether or not the vendor uses effective secure coding practices.
If the software is not as popular, or serves a small niche community, the risk of

undiscovered vulnerabilities is probably higher. In these cases, it pays to look into the certifications of the vendor. A good certification for a software developer is ISO/IEC 27034 Application Security. Unfortunately, you won't find a lot of vendors certified in it. There are also certifications that are very specific to a sector (e.g., ISO 26262 for automotive safety) or a programming language (e.g., ISO/IEC TS 17961:2013 for coding in C) and are a bit less rare to find. Ultimately, however, the security of a vendor's software products is tied to how seriously it takes security in the first place. Absent a secure coding certification, you can look for overall information security management system (ISMS) certifications like ISO/IEC 27001 and FedRAMP, which are difficult to obtain and show that security is taken seriously in an organization.

Open-Source Software

Open-source software is released with a license agreement that allows the user to examine its source code, modify it at will, and even redistribute the modified software (which, per the license, usually requires acknowledgment of the original source and a description of modifications). This may seem perfect, but there are some caveats to keep in mind. First, the software is released as-is, typically without any service or support agreements (though these can be purchased through third parties). This means that your staff may have to

figure out how to install, configure, and maintain the software on their own, unless you contract with someone else to do this for you.

Second, part of the allure of open-source software is that we get access to the source code. This means we can apply all the security tests and assessments we covered earlier. Of course, this only helps if we have the in-house capabilities to examine the source code effectively. Even if we don't, however, we can rely on countless developers and researchers around the world who do examine it (at least for the more popular software). The flip side of that coin, however, is that the adversaries also get to examine the code to either identify vulnerabilities quicker than the defenders or gain insights into how they might

more effectively attack organizations that use specific software.
Perhaps the greatest risk in using open-source software is relying on outdated versions of it. Many of us are used to having software that automatically checks for updates and applies them automatically (either with or without our explicit permission). This is not all that common in open-source software, however, especially libraries. This means we need to develop processes to ensure that all open-source software is periodically updated, possibly in a way that differs from the way in which COTS software is updated.

Third-Party Software
Third-party software, also known as outsourced software, is software made specifically for an organization by a third party. Since the software is custom (or at least customized), it is not considered COTS. Third-party software may rely partly (or even completely) on open-source software, but, having been customized, it may introduce new vulnerabilities.
So, we need a way to verify the security of these products that is probably different from how we would do so with COTS or open-source software.
EXAM TIP Third-party software is custom (or at least customized) to an organization and is not considered commercial off-the-shelf (COTS).

PART VIII

The best (and, sadly, most expensive) way to assess the security of third-party software is to leverage the external or third-party audits discussed in Chapter 18. The way this typically works is that we write into the contract a provision for an external auditor to inspect the software (and possibly the practices through which it was developed), and then issue a report, attesting to the security of the product. Passing this audit can be a condition of finalizing the purchase. Obviously, a sticking point in this negotiation can be who pays for this audit.
Another assessment approach is to arrange for a time-limited trial of the third-party software (perhaps at a nominal cost to the organization), and then have a red team perform an assessment. If you don’t have a red team, you can probably hire one for less money than a formal application security audit would cost. Still, the cost will be considerable, typically (at least) in the low tens of thousands of dollars. As with any other security control, you’d have to balance the cost of the assessment and the loss

you would
incur from insecure software.

Managed Services
As our organizations continue to migrate to cloud services (IaaS, PaaS, and SaaS, discussed in depth in Chapter 7), we should also assess the security impact of those services.
This is highlighted by a 2020 study by global intelligence firm IDC, which found that
nearly 80 percent of the companies surveyed had experienced at least one cloud data
breach in the past 18 months. The top three reasons were misconfigurations, lack of visibility into access settings and activities, and improper access control. The major cloud
services provide tools to help you avoid these pitfalls, but the bottom line is that, if you
don’t have the in-house expertise to secure and assess your cloud services, you really
should consider contracting an expert to help you out.

Chapter Review
Building secure code requires commitment from many parts of the organization, not
just the development and security teams. It starts at the very top with a policy document
that is implemented through standards, procedures, and guidelines. A key part of these is
the inclusion of the various types of tests that must be run regularly (even continuously)
on the software as it is being written, integrated, and prepared for delivery. Software
development environments are complex and could require different approaches from those you’d take in a normal network environment. For this reason, teamwork among all
stakeholders is absolutely critical. A really good way to facilitate this collaboration is by
using the DevSecOps approach introduced in Chapter 24 and highlighted in this one.
Even if your organization doesn’t develop software, it most certainly uses applications
and services developed by others. That’s why the concepts discussed in this chapter are
universally applicable to any cybersecurity leader. You must understand how secure
code is built, so that you can determine whether the software you’re getting from others
presents any undue risks to your organization’s cybersecurity.

Quick Review
• Machine language, which consists of 1’s and 0’s, is the only format that a

computer’s
processor can understand directly and is considered a first-generation language.
• Assembly language is considered a second-generation programming language and uses symbols (called mnemonics) to represent complicated binary codes.
• Third-generation programming languages, such as C/C++, Java, and Python, are known as high-level languages due to their refined programming structures, which allow programmers to leave low-level (system architecture) intricacies to the programming language and focus on their programming objectives.
• Fourth-generation languages (aka very high-level languages) use natural language
processing to allow inexpert programmers to develop code in less time than it would take an experienced software engineer to do so using a third-generation language.

• Fifth-generation programming languages (aka natural languages) approach programming by defining the constraints for achieving a specified result and allowing the development environment to solve problems by itself instead of a programmer having to develop code to deal with individual and specific problems.
• Assemblers are tools that convert assembly language source code into machine code.
• Compilers transform instructions from a source language (high-level) to a target
language (machine), sometimes using an external assembler along the way.
• A garbage collector identifies blocks of memory that were once allocated but are
no longer in use and deallocates the blocks and marks them as free.
• A runtime environment (RTE) functions as a miniature operating system for the program and provides all the resources portable code needs.
• In object-oriented programming (OOP), related functions and data are encapsulated
together in classes, which may then be instantiated as objects.
• Objects in OOP communicate with each other by using messages that conform to the receiving object’s application programming interface (API) definition.
• Cohesion reflects how many different types of tasks a module can carry out, with
the goal being to perform only one task (high cohesion), which makes modules easier to maintain.
• Coupling is a measure of how much a module depends on others; the more dependencies it has, the more complex and difficult the module is to maintain, so we want low (or loose) coupling.
• An API specifies the manner in which a software component interacts with other software components.
• Parameter validation refers to confirming that the parameter values being received
by an application are within defined limits before they are processed by the system.
• A software library is a collection of components that do specific tasks that are
useful to many other components.
• Secure coding is a set of practices that reduce (to acceptable levels) the risk of

vulnerabilities in our software.
• A source code vulnerability is a defect in code that provides a threat actor an
opportunity to compromise the security of a software system.
• Secure coding standards are verifiable, mandatory practices that reduce the risk of
particular types of vulnerabilities in the source code.
• Secure coding guidelines are recommended practices that tend to be less specific
than standards.

PART VIII

• Software-defined security (SDS or SDSec) is a security model in which security functions such as firewalling, IDS/IPS, and network segmentation are implemented in software within an SDN environment.
• Software development tools should be authorized, implemented, and maintained just like any other software product through the organization’s change management
process; developers should not be allowed to install and use arbitrary tools.
• Static application security testing (SAST) is a technique meant to help identify
software defects or security policy violations and is carried out by examining the
source code without executing the program.
• Dynamic application security testing (DAST) refers to the evaluation of a program
in real time, while it is running.
• Fuzzing is a technique used to discover flaws and vulnerabilities in software by
sending large amounts of malformed, unexpected, or random data to the target program in order to trigger failures.
• Continuous integration means that all new code is integrated into the rest of the
system as soon as the developer writes it.
• Continuous delivery is incrementally building a software product that can be released at any time and requires continuous integration.
• A software configuration management (SCM) platform identifies the attributes of software at various points in time and performs a methodical control of changes for the purpose of maintaining software integrity and traceability throughout the SDLC.
• The purpose of a software security assessment is to verify that this entire development process, from organizational policy to delivered product, is working as it should.
• Security assessments of acquired software are essential to mitigate the risk they
could pose to the organization that acquired it.
• The most practical way to assess the security of commercial software is to research
what vulnerabilities and exploits have been discovered by others and decide for ourselves whether or not the vendor uses effective secure coding practices.

• The greatest risk in using open-source software is relying on outdated versions of it.
• The best way to assess the security of third-party (i.e., custom or customized)
software is to perform external or third-party audits.

Questions
Please remember that these questions are formatted and asked in a certain way for a
reason. Keep in mind that the CISSP exam is asking questions at a conceptual level.
Questions may not always have the perfect answer, and the candidate is advised against

always looking for the perfect answer. Instead, the candidate should look for the best
answer in the list.
1. What language is the only one that a computer processor can natively understand
and execute?
A. Machine language
B. Register language
C. Assembly language
D. High-level language

2. To which generation do programming languages such as such as C/C++, Java, and Python belong?
A. Second generation
B. Third generation
C. Fourth generation
D. Fifth generation

3. Which type of tool is specifically designed to convert assembly language into machine language?
A. Compiler
B. Integrated development environment (IDE)
C. Assembler
D. Fuzzer

4. Which of the following is not very useful in assessing the security of acquired
software?
A. The reliability and maturity of the vendor
B. The vendor’s software escrow framework
C. Third-party vulnerability assessments
D. In-house code reviews if source code is available

5. Cohesion and coupling are characteristics of quality code. Which of the following
describes the goals for these two characteristics?
A. Low cohesion, low coupling

B. Low cohesion, high coupling
C. High cohesion, low coupling
D. High cohesion, high coupling

PART VIII

6. Yichen is a new software engineer at Acme Software, Inc. During his first code
review, he is told by his boss that he should use descriptive names for variables in
his code. What is this observation an example of?
A. Secure coding guidelines
B. Secure coding standards
C. Secure software development policy
D. Use of fifth-generation language

7. On what other technology does software-defined security depend?
A. Software-defined storage (SDS)
B. Software-defined networking (SDN)
C. Security orchestration, automation, and response (SOAR)
D. Continuous integration (CI)

8. If you wanted to test source code for vulnerabilities without running it, which
approach would be best?
A. Static application security testing (SAST)
B. Fuzzing
C. Dynamic application security testing (DAST)
D. Manual penetration testing

9. If you wanted to test software for vulnerabilities by executing it and then exposing
it to large amounts of random inputs, which testing technique would you use?
A. Static application security testing (SAST)
B. Fuzzing
C. Dynamic application security testing (DAST)
D. Manual penetration testing

10. Which of the following is not a common reason for data breaches in managed cloud services?
A. Misconfigurations
B. Lack of visibility into access settings and activities
C. Hardware failures
D. Improper access control

Answers
1. A. Machine language, which consists of 1’s and 0’s, is the only format that a computer’s processor can understand directly and is considered a first-generation
language.

1153
2. B. Third-generation programming languages, such as C/C++, Java, and Python, are known as high-level languages due to their refined programming structures, which allow programmers to leave low-level (system architecture) intricacies to the programming language and focus on their programming objectives.
3. C. Assemblers are tools that convert assembly language source code into machine
code. Compilers also generate machine language, but do so by transforming highlevel language code, not assembly language.
4. B. In a software escrow framework, a third party keeps a copy of the source code,
and possibly other materials, which it will release to the customer in specific circumstances such as the developer going out of business. While software escrow is a good business continuity practice, it wouldn’t normally tell us anything about
the security of the software itself. All three other answers are part of a rigorous
assessment of the security of acquired software.
5. C. Cohesion reflects how many different types of tasks a module can carry out, with
the goal being to perform only one task (high cohesion), which makes modules easier to maintain. Coupling is a measure of how much a module depends on others; the more dependencies it has, the more complex and difficult the module is
to maintain, so we want low (or loose) coupling.
6. A. Secure coding guidelines are recommended practices that tend to be less specific
than standards. They might encourage programmers to use variable names that are self-explanatory and to keep functions short (without specifying how short). Secure coding standards, on the other hand, are verifiable, mandatory practices that reduce the risk of particular types of vulnerabilities in the source code.
7. B. Software-defined security (SDS or SDSec) is a security model in which security functions such as firewalling, IDS/IPS, and network segmentation are implemented in software within an SDN environment.
8. A. Static application security testing (SAST) is a technique meant to help identify
software defects or security policy violations and is carried out by examining the
source code without executing the program. All the other answers require that the
code be executed.
9. B. Fuzzing is a technique used to discover flaws and vulnerabilities in software by
sending large amounts of malformed, unexpected, or random data to the target program in order to trigger failures.
10. C. The top three reasons for data breaches in cloud services are misconfigurations,
lack of visibility into access settings and activities, and improper access control.

PART VIII

↟This page intentionally left blank

↟APPENDIX

Comprehensive Questions
Use the following scenario to answer Questions 1–3. Josh has discovered that an organized
hacking ring in China has been targeting his company's research and development department. If these hackers have been able to uncover his company's research findings,
this means they probably have access to his company's intellectual property. Josh thinks
that an e-mail server in his company's DMZ may have been successfully compromised
and a rootkit loaded.
1. Based upon this scenario, what is most likely the biggest risk Josh's company needs to be concerned with?
A. Market share drop if the attackers are able to bring the specific product to

market more quickly than Josh's company.
B. Confidentiality of e-mail messages. Attackers may post all captured e-mail messages to the Internet.
C. Impact on reputation if the customer base finds out about the attack.
D. Depth of infiltration of attackers. If attackers have compromised other systems, more confidential data could be at risk.
2. The attackers in this situation would be seen as which of the following?
A. Vulnerability
B. Threat
C. Risk
D. Threat agent
3. If Josh is correct in his assumptions, which of the following best describes the
vulnerability, threat, and exposure, respectively?
A. E-mail server is hardened, an entity could exploit programming code flaw, server is compromised and leaking data.
B. E-mail server is not patched, an entity could exploit a vulnerability, server is
hardened.
C. E-mail server misconfiguration, an entity could exploit misconfiguration, server is compromised and leaking data.
D. DMZ firewall misconfiguration, an entity could exploit misconfiguration, internal e-mail server is compromised.

1155

A

↟CISSP All-in-One Exam Guide

1156
4. Aaron is a security manager who needs to develop a solution to allow his company's
mobile devices to be authenticated in a standardized and centralized manner using

digital certificates. The applications these mobile clients use require a TCP connection. Which of the following is the best solution for Aaron to implement?
A. TACACS+
B. RADIUS
C. Diameter
D. Mobile IP

5. Terry is a security manager for a credit card processing company. His company uses
internal DNS servers, which are placed within the LAN, and external DNS servers, which are placed in the DMZ. The company also relies on DNS servers provided by its service provider. Terry has found out that attackers have been able to manipulate
several DNS server caches to point employee traffic to malicious websites. Which of
the following best describes the solution this company should implement?
A. IPSec
B. PKI
C. DNSSEC
D. MAC-based security

6. Which of the following is not a key provision of the GDPR?
A. Requirement for consent from data subjects
B. Right to be informed
C. Exclusion for temporary workers
D. Right to be forgotten

7. Jane is suspicious that an employee is sending sensitive data to one of the company’s
competitors but is unable to confirm this. The employee has to use this data for daily activities, thus it is difficult to properly restrict the employee’s access rights.
In this scenario, which best describes the company’s vulnerability, threat, risk,
and necessary control?
A. Vulnerability is employee access rights, threat is internal entities misusing

privileged access, risk is the business impact of data loss, and the necessary control is detailed network traffic monitoring.
B. Vulnerability is lack of user monitoring, threat is internal entities misusing
privileged access, risk is the business impact of data loss, and the necessary control is detailed user activity logs.
C. Vulnerability is employee access rights, threat is internal employees misusing
privileged access, risk is the business impact of confidentiality, and the necessary control is multifactor authentication.
D. Vulnerability is employee access rights, threat is internal users misusing privileged access, risk is the business impact of confidentiality, and the necessary control is CCTV.

8. Which of the following best describes what role-based access control offers organizations in reducing administrative burdens?
A. It allows entities closer to the resources to make decisions about who can and

cannot access resources.
B. It provides a centralized approach for access control, which frees up department managers.
C. User membership in roles can be easily revoked and new ones established as job assignments dictate.
D. It enforces an enterprise-wide security policy, standards, and guidelines.
9. Mark works for a large corporation operating in multiple countries worldwide. He is reviewing his company's policies and procedures dealing with data breaches.
Which of the following is an issue that he must take into consideration?
A. Each country may or may not have unique notification requirements.
B. All breaches must be announced to affected parties within 24 hours.
C. Breach notification is a "best effort" process and not a guaranteed process.
D. Breach notifications are avoidable if all PII is removed from data stores.
10. A software development company released a product that committed several errors that were not expected once deployed in their customers' environments. All of the software code went through a long list of tests before being released.
The team manager found out that after a small change was made to the code, the program was not tested before it was released. Which of the following tests was most likely not conducted?
A. Unit
B. Compiled
C. Integration
D. Regression
11. Which of the following should not be considered as part of the supply chain risk
management process for a smartphone manufacturer?
A. Hardware Trojans inserted by downstream partners
B. ISO/IEC 27001
C. Hardware Trojans inserted by upstream partners
D. NIST Special Publication 800-161
12. Data sovereignty is increasingly becoming an issue that most of us in cybersecurity
should address within our organizations. What does the term data sovereignty mean?
A. Certain types of data concerning a country's citizens must be stored and processed in that country.
B. Data on a country's citizens must be stored and processed according to that country's laws, regardless of where the storing/processing takes place.

C. Certain types of data concerning a country's citizens are the sovereign

property of that data subject.
D. Data on a country's citizens must never cross the sovereign borders of another

country.
Use the following scenario to answer Questions 13–15. Jack has just been hired as the security officer for a large hospital system. The organization develops some of its own proprietary applications. The organization does not have as many layers of controls when it comes to the data processed by these applications, since it is assumed that external entities will not understand the internal logic of the applications. One of the first things that Jack wants to carry out is a risk assessment to determine the organization's current risk profile. He also tells his boss that the hospital should become ISO certified to bolster its customers' and partners' confidence in its risk management processes.

13. Which of the following approaches has been implemented in this scenario?
A. Defense-in-depth
B. Security through obscurity
C. Information security management system
D. ISO/IEC 27001

14. Which ISO/IEC standard would be best for Jack to follow to meet his goals?
A. ISO/IEC 27001
B. ISO/IEC 27004
C. ISO/IEC 27005
D. ISO/IEC 27006

15. Which standard should Jack suggest to his boss for compliance with best practices regarding storing and processing sensitive medical information?
A. ISO/IEC 27004
B. ISO/IEC 27001
C. ISO/IEC 27799
D. ISO/IEC 27006

16. You just received an e-mail from one of your hardware manufacturers notifying you that it will no longer manufacture a certain product and, after the end of the year, you won't be able to send it in for repairs, buy spare parts, or get technical assistance from that manufacturer. What term describes this?
A. End-of-support (EOS)
B. End-of-service-life (EOSL)
C. Deprecation
D. End-of-life (EOL)

↟Appendix A: Comprehensive Questions

17. The confidentiality of sensitive data is protected in different ways depending on the state of the data. Which of the following is the best approach to protecting

data in transit?
A. SSL
B. VPN
C. IEEE 802.1X
D. Whole-disk encryption

18. Your boss asks you to put together a report describing probable adverse effects on
your assets caused by specific threat sources. What term describes this?
A. Risk analysis
B. Threat modeling
C. Attack trees
D. MITRE ATT&CK

19. A(n) _____ is the graphical representation of data commonly used on websites. It is a skewed representation of characteristics a person must enter to prove
that the subject is a human and not an automated tool, as in a software robot.
A. anti-spoofing symbol
B. CAPTCHA
C. spam anti-spoofing symbol
D. CAPCHAT

20. Mark has been asked to interview individuals to fulfill a new position in his company, chief privacy officer (CPO). What is the function of this type of position?
A. Ensuring that company financial information is correct and secure
B. Ensuring that customer, company, and employee data is protected
C. Ensuring that security policies are defined and enforced
D. Ensuring that partner information is kept safe

21. A risk management program must be developed properly and in the right sequence.
Which of the following provides the correct sequence for the steps listed?
i. Develop a risk management team.
ii. Calculate the value of each asset.
iii. Identify the vulnerabilities and threats that can affect the identified assets.
iv. Identify company assets to be assessed.
A. i, iii, ii, iv
B. ii, i, iv, iii
C. iii, i, iv, ii
D. i, iv, ii, iii

22. Juan needs to assess the performance of a critical web application that his company recently upgraded. Some of the new features are very profitable, but not frequently used. He wants to ensure that the user experience is positive, but
doesn’t want to wait for the users to report problems. Which of the following techniques should Juan use?
A. Real user monitoring

B. Synthetic transactions
C. Log reviews
D. Management review

23. Which of the following best describes a technical control for dealing with the
risks presented by data remanence?
A. Encryption
B. Data retention policies
C. File deletion
D. Using solid-state drives (SSDs)

24. George is the security manager of a large bank, which provides online banking
and other online services to its customers. George has recently found out that some of the bank's customers have complained about changes to their bank accounts that they did not make. George worked with the security team and found out that all changes took place after proper authentication steps were completed. Which of the following describes what most likely took place in this situation?
A. Web servers were compromised through cross-scripting attacks.
B. TLS connections were decrypted through a man-in-the-middle attack.
C. Personal computers were compromised with malware that installed

keyloggers.
D. Web servers were compromised and masquerading attacks were carried out.
25. Internet Protocol Security (IPSec) is actually a suite of protocols. Each protocol
within the suite provides different functionality. Which of the following is not a
function or characteristic of IPSec?
A. Encryption
B. Link layer protection
C. Authentication
D. Protection of packet payloads and the headers

26. In what order would a typical PKI perform the following transactions?
i. Receiver decrypts and obtains session key.
ii. Public key is verified.
iii. Public key is sent from a public directory.
iv. Sender sends a session key encrypted with receiver's public key.
A. iv, iii, ii, i
B. ii, i, iii, iv
C. iii, ii, iv, i
D. ii, iv, iii, i

Use the following scenario to answer Questions 27–28. Tim is the CISO for a large
distributed financial investment organization. The company's network is made up of
different network devices and software applications, which generate their own

proprietary
logs and audit data. Tim and his security team have become overwhelmed with trying
to review all of the log files when attempting to identify if anything suspicious is taking
place within the network. Another issue Tim’s team needs to deal with is that many of
the network devices have automated IPv6-to-IPv4 tunneling enabled by default, which
is not what the organization needs.
27. Which of the following is the best solution to Tim’s difficulties handling the
quantity and diversity of logs and audit data?
A. Event correlation tools
B. Intrusion detection systems
C. Security information and event management
D. Hire more analysts

28. How could Tim best address the IP version issue described in the scenario?
A. Change management
B. Zero trust
C. Converged protocols
D. Configuration management

29. Which of the following is not a concern of a security professional considering
adoption of Internet of Things (IoT) devices?
A. Weak or nonexistent authentication mechanisms
B. Vulnerability of data at rest and data in motion
C. Difficulty of deploying patches and updates
D. High costs associated with connectivity

30. What is an advantage of microservices compared to traditional server-based architectures?
A. Web services support
B. Security
C. Scalability
D. Database connectivity

31. _____, a declarative access control policy language implemented in XML and a processing model, describes how to interpret security policies. _____ is an XML-based language that allows for the exchange of provisioning data between applications, which could reside in one organization or many.
A. Service Provisioning Markup Language (SPML), Extensible Access Control

Markup Language (XACML)
B. Extensible Access Control Markup Language (XACML), Service Provisioning Markup Language (SPML)
C. Extensible Access Control Markup Language (XACML), Security Assertion Markup Language (SAML)
D. Security Assertion Markup Language (SAML), Service Provisioning Markup

Language (SPML)
32. Doors configured in fail-safe mode assume what position in the event of a power
failure?
A. Open and locked
B. Closed and locked
C. Closed and unlocked
D. Open
33. Next-generation firewalls combine the best attributes of other types of firewalls.
Which of the following is not a common characteristic of these firewall types?
A. Integrated intrusion prevention system
B. Sharing signatures with cloud-based aggregators
C. Automated incident response
D. High cost
34. The purpose of security awareness training is to expose personnel to security issues
so that they may be able to recognize them and better respond to them. Which of the following is not normally a topic covered in security awareness training?
A. Social engineering
B. Phishing
C. Whaling
D. Trolling

Use the following scenario to answer Questions 35–36. Zack is a security consultant who
has been hired to help an accounting company improve some of its current e-mail security practices. The company wants to ensure that when its clients send the company
accounting files and data, the clients cannot later deny sending these messages. The
company also wants to integrate a more granular and secure authentication method for
its current mail server and clients.
35. Which of the following best describes how client messages can be dealt with and
addresses the first issue outlined in the scenario?
A. The company needs to integrate a public key infrastructure and the Diameter

protocol.
B. The company needs to require that clients encrypt messages with their public key before sending them to the company.
C. The company needs to have all clients sign a formal document outlining nonrepudiation requirements.
D. The company needs to require that clients digitally sign messages that contain
financial information.
36. Which of the following would be the best solution to integrate to meet the authentication requirements outlined in the scenario?
A. TLS
B. IPSec

C. 802.1X
D. SASL
37. Which of the following is not considered a secure coding practice?
A. Validate user inputs
B. Default deny
C. Defense in depth
D. High (tight) coupling
38. A _____ is the amount of time it should take to recover from a disaster,
and a _____ is the amount of data, measured in time, that can be lost and be tolerable from that same event.
A. recovery time objective, recovery point objective
B. recovery point objective, recovery time objective
C. maximum tolerable downtime, work recovery time
D. work recovery time, maximum tolerable downtime

39. Mary is doing online research about prospective employers and discovers a way to
compromise a small company's personnel files. She decides to take a look around, but does not steal any information. Is she still committing a crime even if she does not steal any of the information?
A. No, since she does not steal any information, she is not committing a crime.
B. Probably, because she has gained unauthorized access.
C. Not if she discloses the vulnerability she exploited to the company.
D. Yes, she could jeopardize the system without knowing it.

40. In the structure of Extensible Access Control Markup Language (XACML), a Subject element is the _____, a Resource element is the _____, and an Action element is the _____.
A. requesting entity, requested entity, types of access
B. requested entity, requesting entity, types of access
C. requesting entity, requested entity, access control
D. requested entity, requesting entity, access control

41. The Mobile IP protocol allows location-independent routing of IP datagrams on
the Internet. Each mobile node is identified by its _____, disregarding its current location in the Internet. While away from its home network, a mobile node is associated with a _____.
A. prime address, care-of address
B. home address, care-of address
C. home address, secondary address
D. prime address, secondary address

42. Because she has many different types of security products and solutions, Joan
wants to purchase a product that integrates her many technologies into one user interface. She would like her staff to analyze all security alerts from the same application environment. Which of the following would best fit Joan's needs?
A. Dedicated appliance
B. Data analytics platform

C. Hybrid IDS\IPS integration
D. Security information and event management (SIEM)

43. When classifying an information asset, which of the following is true concerning
its sensitivity?
A. It is commensurate with how its loss would impact the fundamental business

processes of the organization.
B. It is determined by its replacement cost.

C. It is determined by the product of its replacement cost and the probability of

its compromise.
D. It is commensurate with the losses to an organization if it were revealed to unauthorized individuals.
44. Which of the following is an international organization that helps different governments come together and tackle the economic, social, and governance challenges of a globalized economy and provides guidelines on the protection of privacy and transborder flows of personal data rules?
A. Council of Global Convention on Cybercrime
B. Council of Europe Convention on Cybercrime
C. Organisation for Economic Co-operation and Development
D. Organisation for Cybercrime Co-operation and Development
45. System ports allow different computers to communicate with each other's services
and protocols. The Internet Assigned Numbers Authority (IANA) has assigned registered ports to be _____ and dynamic ports to be _____.
A. 0–1024, 49152–65535
B. 1024–49151, 49152–65535
C. 1024–49152, 49153–65535
D. 0–1024, 1025–49151
46. When conducting a quantitative risk analysis, items are gathered and assigned
numeric values so that cost/benefit analysis can be carried out. Which of the following formulas could be used to understand the value of a safeguard?
A. (ALE before implementing safeguard) – (ALE after implementing safeguard) – (annual cost of safeguard) = value of safeguard to the organization
B. (ALE before implementing safeguard) – (ALE during implementing safeguard) – (annual cost of safeguard) = value of safeguard to the organization
C. (ALE before implementing safeguard) – (ALE while implementing safeguard) – (annual cost of safeguard) = value of safeguard to the organization
D. (ALE before implementing safeguard) – (ALE after implementing safeguard) – (annual cost of asset) = value of safeguard to the organization
47. Patty is giving a presentation next week to the executive staff of her company. She
wants to illustrate the benefits of the company using specific cloud computing solutions. Which of the following does not properly describe one of these benefits
or advantages?

A. Organizations have more flexibility and agility in IT growth and functionality.
B. Cost of computing can be increased since it is a shared delivery model.

C. Location independence can be achieved because the computing is not centralized and tied to a physical data center.
D. Scalability and elasticity of resources can be accomplished in near real-time through automation.

Use the following scenario to answer Questions 48–49. Francisca is the new manager of the in-house software designers and programmers. She has been telling her team that before design and programming on a new product begins, a formal architecture needs to be developed. She also needs this team to understand security issues as they pertain to software design. Francisca has shown the team how to follow a systematic approach that allows them to understand different ways in which the software products they develop could be compromised by specific threat actors.

48. Which of the following best describes what an architecture is in the context of this scenario?
A. Tool used to conceptually understand the structure and behavior of a complex entity through different views
B. Formal description and representation of a system and the components that make it up
C. Framework used to create individual architectures with specific views
D. Framework that is necessary to identify needs and meet all of the stakeholder requirements

49. Which of the following best describes the approach Francisca has shown her team as outlined in the scenario?
A. Attack surface analysis
B. Threat modeling
C. Penetration testing
D. Double-blind penetration testing

50. Barry was told that the IDS product that is being used on the network has heuristic capabilities. Which of the following best describes this functionality?
A. Gathers packets and reassembles the fragments before assigning anomaly values
B. Gathers data and assesses the likelihood of it being malicious in nature
C. Gathers packets and compares their payload values to a signature engine
D. Gathers packet headers to determine if something suspicious is taking place within the network traffic

1167
51. Bringing in third-party auditors has advantages over using an internal team. Which of the following is not true about using external auditors?
A. They are required by certain governmental regulations.
B. They bring experience gained by working in many other organizations.
C. They know the organization’s processes and technology better than anyone else.
D. They are less influenced by internal culture and politics.

52. Don is a senior manager of an architectural firm. He has just found out that a key
contract was renewed, allowing the company to continue developing an operating system that was idle for several months. Excited to get started, Don begins work on the operating system privately, but cannot tell his staff until the news is announced publicly in a few days. However, as Don begins making changes in the software, various staff members notice changes in their connected systems, even though they have a lower security level than Don. What kind of model could be used to ensure this does not happen?
A. Biba
B. Bell-LaPadula
C. Noninterference
D. Clark-Wilson

53. Betty has received several e-mail messages from unknown sources that try and entice her to click a specific link using a “Click Here” approach. Which of the following best describes what is most likely taking place in this situation?
A. DNS pharming attack
B. Embedded hyperlink is obfuscated
C. Malware back-door installation
D. Bidirectional injection attack

54. Rebecca is an internal auditor for a large retail company. The company has a number of web applications that run critical business processes with customers and partners around the world. Her company would like to ensure the security of technical controls on these processes. Which of the following would not be a good approach to auditing these technical controls?
A. Log reviews
B. Code reviews
C. Personnel background checks
D. Misuse case testing

1168
55. Which of the following multiplexing technologies analyzes statistics related to
the typical workload of each input device and makes real-time decisions on how much time each device should be allocated for data transmission?
A. Time-division multiplexing
B. Wave-division multiplexing
C. Frequency-division multiplexing
D. Statistical time-division multiplexing

56. In a VoIP environment, the Real-time Transport Protocol (RTP) and RTP

Control
Protocol (RTCP) are commonly used. Which of the following best describes the difference between these two protocols?
A. RTCP provides a standardized packet format for delivering audio and video

over IP networks. RTP provides out-of-band statistics and control information to provide feedback on QoS levels.
B. RTP provides a standardized packet format for delivering data over IP networks.
RTCP provides control information to provide feedback on QoS levels.
C. RTP provides a standardized packet format for delivering audio and video over MPLS networks. RTCP provides control information to provide feedback on
QoS levels.
D. RTP provides a standardized packet format for delivering audio and video over IP networks. RTCP provides out-of-band statistics and control information to provide feedback on QoS levels.
57. Which of the following is not descriptive of an edge computing architecture?
A. It eliminates the need for cloud infrastructure.
B. Processing and storage assets are close to where they’re needed.
C. It reduces latency and network traffic.
D. It typically has three layers.
58. Which cryptanalytic attack method is characterized by the identification of statistically significant patterns in the ciphertext generated by a cryptosystem?
A. Differential attack
B. Implementation attack
C. Frequency analysis
D. Side-channel attack
59. IPSec’s main protocols are AH and ESP. Which of the following services does AH provide?
A. Confidentiality and authentication
B. Confidentiality and availability
C. Integrity and accessibility
D. Integrity and authentication

60. When multiple databases exchange transactions, each database is updated. This
can happen many times and in many different ways. To protect the integrity of the data, databases should incorporate a concept known as an ACID test. What does this acronym stand for?
A. Availability, confidentiality, integrity, durability
B. Availability, consistency, integrity, durability
C. Atomicity, confidentiality, isolation, durability
D. Atomicity, consistency, isolation, durability

Use the following scenario to answer Questions 61–63. Jim works for a large energy company.
His senior management just conducted a meeting with Jim’s team with the purpose of reducing IT costs without degrading their security posture. The senior management
decided to move all administrative systems to a cloud provider. These systems

are
proprietary applications currently running on Linux servers.
61. Which of the following services would allow Jim to transition all administrative
custom applications to the cloud while leveraging the service provider for security
and patching of the cloud platforms?
A. IaaS
B. PaaS
C. SaaS
D. IDaaS

62. Which of the following would not be an issue that Jim would have to consider in
transitioning administrative services to the cloud?
A. Privacy and data breach laws in the country where the cloud servers are

located
B. Loss of efficiencies, performance, reliability, scalability, and security
C. Security provisions in the terms of service
D. Total cost of ownership compared to the current systems
63. Which of the following secure design principles would be most important to consider as Jim plans the transition to the cloud?
A. Defense in depth
B. Secure defaults
C. Shared responsibility
D. Zero trust

64. A group of software designers are at a stage in their software development project
where they need to reduce the amount of code running, reduce entry points available to untrusted users, reduce privilege levels as much as possible, and eliminate unnecessary services. Which of the following best describes the first step
the team needs to carry out to accomplish these tasks?
A. Attack surface analysis
B. Software development life cycle
C. Risk assessment
D. Unit testing

65. Jenny needs to engage a new software development company to create her company’s
internal banking software. The software needs to be created specifically for her company’s environment, so it must be proprietary in nature. Which of the following
would be useful for Jenny to use as a gauge to determine how advanced the various
software development companies are in their processes?
A. Waterfall methodology
B. Capability Maturity Model Integration level
C. Auditing results

D. Key performance metrics

66. Which type of organization would be likeliest to implement Virtual eXtensible
Local Area Network (VxLAN) technology?
A. Organizations that need to support more than 2,048 VLANs
B. Small and medium businesses
C. Organizations with hosts in close proximity to each other
D. Cloud service providers with hundreds of customers

67. Kerberos is a commonly used access control and authentication technology. It is
important to understand what the technology can and cannot do and its potential downfalls. Which of the following is not a potential security issue that must be addressed when using Kerberos?
i. The KDC can be a single point of failure.
ii. The KDC must be scalable.
iii. Secret keys are temporarily stored on the users' workstations.
iv. Kerberos is vulnerable to password guessing.
A. i, iv
B. iii
C. All of them
D. None of them

1171
68. If the annualized loss expectancy (ALE) for a specific asset is $100,000, and after
implementation of a control to safeguard the asset the new ALE is $45,000 and the
annual cost of the control is $30,000, should the company implement this control?
A. Yes
B. No
C. Not enough information
D. Depends on the annualized rate of occurrence (ARO)

69. ISO/IEC 27000 is a growing family of ISO/IEC information security management system (ISMS) standards. Which of the following provides an incorrect mapping of the individual standard number to its description?
A. ISO/IEC 27002: Code of practice for information security controls
B. ISO/IEC 27003: ISMS implementation guidance
C. ISO/IEC 27004: ISMS monitoring, measurement, analysis, and evaluation
D. ISO/IEC 27005: ISMS auditing guidelines

70. Yazan leads the IT help desk at a large manufacturing company. He is concerned
about the amount of time his team spends resetting passwords for the various accounts that each of his organizational users has. All of the following would be
good approaches to alleviating this help desk load except which one?
A. Single sign-on (SSO)
B. Just-in-time (JIT) access

C. Password managers
D. Self-service password reset

71. Encryption and decryption can take place at different layers of an operating system, application, and network stack. End-to-end encryption happens within the _____. IPSec encryption takes place at the _____ layer. PPTP encryption takes place at the _____ layer. Link encryption takes place at the _____ and _____ layers.
A. applications, transport, data link, data link, physical
B. applications, transport, network, data link, physical
C. applications, network, data link, data link, physical
D. network, transport, data link, data link, physical

72. Which of the following best describes the difference between hierarchical storage
management (HSM) and storage area network (SAN) technologies?
A. HSM uses optical or tape jukeboxes, and SAN is a network of connected

storage systems.
B. SAN uses optical or tape jukeboxes, and HSM is a network of connected storage systems.

C. HSM and SAN are one and the same. The difference is in the implementation.
D. HSM uses optical or tape jukeboxes, and SAN is a standard of how to develop

and implement this technology.
73. Which legal system is characterized by its reliance on previous interpretations of
the law?
A. Tort
B. Customary
C. Common
D. Civil (code)
74. In order to be admissible in court, evidence should normally be which of the following?
A. Subpoenaed
B. Relevant
C. Motioned
D. Adjudicated
75. Which type of authorization mechanism can incorporate historical data into its
access control decision-making in real time?
A. Rule-based access control
B. Risk-based access control
C. Attribute-based access control
D. Discretionary access control
76. Which of the following is an XML-based protocol that defines the schema of how web service communication takes place over HTTP transmissions?
A. Service-Oriented Protocol
B. Active X Protocol
C. SOAP

D. Web Ontology Language
77. Which of the following has an incorrect definition mapping?
i. Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE) Team-oriented approach that assesses organizational and IT risks through facilitated workshops
ii. Facilitated Risk Analysis Process (FRAP) Stresses prescreening activities so that the risk assessment steps are only carried out on the item(s) that need(s) it the most
iii. ISO/IEC 27005 International standard for the implementation of a risk management program that integrates into an information security management system (ISMS)

iv. Failure Modes and Effect Analysis (FMEA)

Approach that dissects a component into its basic functions to identify flaws and those flaws' effects
v. Fault tree analysis Approach to map specific flaws to root causes in complex systems
A. None of them
B. ii
C. iii, iv
D. v
78. For an enterprise security architecture to be successful in its development and implementation, which of the following items must be understood and followed?
i. Strategic alignment
ii. Process enhancement
iii. Business enablement
iv. Security effectiveness
A. i, ii
B. ii, iii
C. i, ii, iii, iv
D. iii, iv
79. Which of the following best describes the purpose of the Organisation for Economic Co-operation and Development (OECD)?
A. An international organization where member countries come together and tackle the economic, social, and governance challenges of a globalized economy
B. A national organization that helps different governments come together and tackle the economic, social, and governance challenges of a globalized economy
C. A United Nations body that regulates economic, social, and governance issues of a globalized economy
D. A national organization that helps different organizations come together and tackle the economic, social, and governance challenges of a globalized economy
80. Many enterprise architecture models have been developed over the years for specific purposes. Some of them can be used to provide structure for information security processes and technology to be integrated throughout an organization. Which of the following provides an incorrect mapping between the architecture type and the associated definition?
A. Zachman Framework Model and methodology for the development of information security enterprise architectures

B. TOGAF Model and methodology for the development of enterprise architectures developed by The Open Group

U.S. Department of Defense architecture framework that ensures interoperability of systems to meet military mission goals

D. SABSA Framework and methodology for enterprise security architecture and service management

81. Which of the following best describes the difference between the role of the ISO/IEC 27000 series and COBIT?

A. COBIT provides a high-level overview of security program requirements, while the ISO/IEC 27000 series provides the objectives of the individual security controls.

B. The ISO/IEC 27000 series provides a high-level overview of security program requirements, while COBIT maps IT goals to enterprise goals to stakeholder needs.

C. COBIT is process oriented, and the ISO/IEC 27000 series is solution oriented.

D. The ISO/IEC 27000 series is process oriented, and COBIT is solution oriented.

82. The Capability Maturity Model Integration (CMMI) approach is being used more frequently in security program and enterprise development. Which of the following provides an incorrect characteristic of this model?

A. It provides a pathway for how incremental improvement can take place.

B. It provides structured steps that can be followed so an organization can evolve from one level to the next and constantly improve its processes.

C. It was created for process improvement and developed by Carnegie Mellon.

D. It was built upon the SABSA model.

83. If Jose wanted to use a risk assessment methodology across the entire organization and allow the various business owners to identify risks and know how to deal with them, what methodology would he use?

A. Qualitative

B. COBIT

C. FRAP

D. OCTAVE

84. Information security is a field that is maturing and becoming more organized and standardized. Organizational security models should be based on an enterprise architecture framework. Which of the following best describes what an enterprise architecture framework is and why it would be used?

A. Mathematical model that defines the secure states that various software components can enter and still provide the necessary protection

B. Conceptual model that is organized into multiple views addressing each of the stakeholder’s concerns

C. DoDAF

C. Business enterprise framework that is broken down into six conceptual levels to ensure security is deployed and managed in a controllable manner

D. Enterprise framework that allows for proper security governance
85. Which of the following provides a true characteristic of a fault tree analysis?
A. Fault trees are assigned qualitative values to faults that can take place over a
series of business processes.
B. Fault trees are assigned failure mode values.
C. Fault trees are labeled with actual numbers pertaining to failure probabilities.
D. Fault trees are used in a stepwise approach to software debugging.
86. It is important that organizations ensure that their security efforts are effective
and measurable. Which of the following is not a common method used to track the effectiveness of security efforts?
A. Service level agreement
B. Return on investment
C. Balanced scorecard system
D. Provisioning system
87. Capability Maturity Model Integration (CMMI) is a process improvement approach that is used to help organizations improve their performance. The CMMI model may also be used as a framework for appraising the process maturity of the organization. Which of the following is an incorrect mapping of the levels
that may be assigned to an organization based upon this model?
i. Maturity Level 2 – Managed or Repeatable
ii. Maturity Level 3 – Defined
iii. Maturity Level 4 – Quantitatively Managed
iv. Maturity Level 5 – Optimizing
A. i
B. i, ii
C. All of them
D. None of them
88. An organization’s information systems risk management (ISRM) policy should address many items to provide clear direction and structure. Which of the following is not a core item that should be covered in this type of policy?
i. The objectives of the ISRM team
ii. The level of risk the organization will accept and what is considered an acceptable level of risk
iii. Formal processes of risk identification

iv. The connection between the ISRM policy and the organization’s strategic

planning processes
v. Responsibilities that fall under ISRM and the roles to fulfill them
vi. The mapping of risk to specific physical controls
vii. The approach toward changing staff behaviors and resource allocation in response to risk analysis
viii. The mapping of risks to performance targets and budgets
ix. Key metrics and performance indicators to monitor the effectiveness of controls
A. ii, v, ix

B. vi
C. v
D. vii, ix
89. More organizations are outsourcing supporting functions to allow them to focus on their core business functions. Organizations use hosting companies to maintain websites and e-mail servers, service providers for various telecommunication connections, disaster recovery companies for co-location capabilities, cloud computing providers for infrastructure or application services,
developers for software creation, and security companies to carry out vulnerability
management. Which of the following items should be included during the analysis of an outsourced partner or vendor?
i. Conduct onsite inspection and interviews
ii. Review contracts to ensure security and protection levels are agreed upon
iii. Ensure service level agreements are in place
iv. Review internal and external audit reports and third-party reviews
v. Review references and communicate with former and existing customers
A. ii, iii, iv
B. iv, v
C. All of them
D. i, ii, iii
90. Which of the following is normally not an element of e-discovery?
A. Identification
B. Preservation
C. Production
D. Remanence

1177
91. A financial institution has developed its internal security program based upon the
ISO/IEC 27000 series. The security officer has been told that metrics need to be developed and integrated into this program so that effectiveness can be gauged. Which of the following standards should be followed to provide this type of guidance and functionality?
A. ISO/IEC 27002
B. ISO/IEC 27003
C. ISO/IEC 27004
D. ISO/IEC 27005

92. Which of the following is not an advantage of using content distribution networks?
A. Improved responsiveness to regional users
B. Resistance to ARP spoofing attacks
C. Customization of content for regional users
D. Resistance to DDoS attacks

93. Sana has been asked to install a cloud access security broker (CASB) product for
her company’s environment. What is the best description for what CASBs are commonly used for?
A. Monitor end-user behavior and enforce policies across cloud services

B. Provision secure cloud services
C. Enforce access controls to cloud services through X.500 databases
D. Protect cloud services from certain types of attacks

94. Which of the following allows a user to be authenticated across multiple IT systems and enterprises?
A. Single sign-on (SSO)
B. Session management
C. Federated identity
D. Role-based access control (RBAC)

95. Which of the following is a true statement pertaining to markup languages?
A. Hypertext Markup Language (HTML) came from Generalized Markup

Language (GML), which came from Standard Generalized Markup Language (SGML).
B. Hypertext Markup Language (HTML) came from Standard Generalized Markup Language (SGML), which came from Generalized Markup Language (GML).

C. Standard Generalized Markup Language (SGML) came from Hypertext Markup

Language (HTML), which came from Generalized Markup Language (GML).
D. Standard Generalized Markup Language (SGML) came from Generalized Markup Language (GML), which came from Hypertext Markup Language (HTML).
96. What is Extensible Markup Language (XML) and why was it created?
A. A specification that provides a structure for creating other markup languages and still allow for interoperability
B. A specification that is used to create static and dynamic websites
C. A specification that outlines a detailed markup language dictating all formats
of all companies that use it
D. A specification that does not allow for interoperability for the sake of security
97. Which access control policy is based on the necessary operations and tasks users
need to fulfill their responsibilities within an organization and allows for implicit
permission inheritance using a nondiscretionary model?
A. Rule-based
B. Role-based
C. Identity-based
D. Mandatory
98. Which of the following centralized access control protocols would a security professional choose if her network consisted of multiple protocols, including Mobile IP, and had users connecting via wireless and wired transmissions?
A. RADIUS
B. TACACS+
C. Diameter
D. Kerberos

99. Javad is the security administrator at a credit card processing company. The company has many identity stores, which are not properly synchronized. Javad is going to oversee the process of centralizing and synchronizing the identity data within the company. He has determined that the data in the HR database will be considered the most up-to-date data, which cannot be overwritten by the software in other identity stores during their synchronization processes. Which of
the following best describes the role of this database in the identity management
structure of the company?
A. Authoritative system of record
B. Infrastructure source server
C. Primary identity store
D. Hierarchical database primary

100. Proper access control requires a structured user provisioning process. Which of
the following best describes user provisioning?
A. The creation, maintenance, and deactivation of user objects and attributes as

they exist in one or more systems, directories, or applications, in response to business processes
B. The creation, maintenance, activation, and delegation of user objects and attributes as they exist in one or more systems, directories, or applications, in
response to compliance processes
C. The maintenance of user objects and attributes as they exist in one or more systems, directories, or applications, in response to business processes
D. The creation and deactivation of user objects and attributes as they exist in one
or more systems, directories, or applications, in response to business processes
101. Which of the following protocols would an Identity as a Service (IDaaS) provider
use to authenticate you to a third party?
A. Diameter
B. OAuth
C. Kerberos
D. OpenID Connect
102. Johana needs to ensure that her company’s application can accept provisioning
data from the company’s partner’s application in a standardized method. Which of the following best describes the technology that Johana should implement?
A. Service Provisioning Markup Language
B. Extensible Provisioning Markup Language
C. Security Assertion Markup Language
D. Security Provisioning Markup Language
103. Lynn logs into a website and purchases an airline ticket for her upcoming trip.
The website also offers her pricing and package deals for hotel rooms and rental cars while she is completing her purchase. The airline, hotel, and rental companies are all separate and individual companies. Lynn decides to purchase

her hotel room through the same website at the same time. The website is using Security Assertion Markup Language to allow for this type of federated identity management functionality. In this example which entity is the principal,
which entity is the identity provider, and which entity is the service provider, respectively?
A. Portal, Lynn, hotel company
B. Lynn, airline company, hotel company
C. Lynn, hotel company, airline company
D. Portal, Lynn, airline company

104. John is the new director of software development within his company. Several
proprietary applications offer individual services to the employees, but the employees have to log into each and every application independently to gain access
to these discrete services. John would like to provide a way that allows each of the
services provided by the various applications to be centrally accessed and controlled.
Which of the following best describes the architecture that John should deploy?
A. Service-oriented architecture
B. Web services architecture
C. Single sign-on architecture
D. Hierarchical service architecture

105. Which security model is defined by three main rules: simple security, star property, and strong star property?
A. Biba
B. Bell-LaPadula
C. Brewer-Nash
D. Noninterference

106. Khadijah is leading a software development team for her company. She knows the importance of conducting an attack surface analysis and developing a threat model. During which phase of the software development life cycle should she perform these actions?
A. Requirements gathering
B. Testing and validation
C. Release and maintenance
D. Design

107. Bartosz is developing a new web application for his marketing department. One
of the requirements for the software is that it allows users to post specific content
to LinkedIn and Twitter directly from the web app. Which technology would allow him to do this?
A. OpenID Connect
B. OAuth
C. SSO

D. Federated Identity Management

108. Applications may not work on systems with specific processors. Which of the following best describes why an application may work on an Intel processor but not on an AMD processor?
A. The application was not compiled to machine language that is compatible

with the AMD architecture.
B. It is not possible for the same application to run on both Intel and AMD processors.

C. The application was not compiled to machine language that is compatible

with the Windows architecture.
D. Only applications written in high-level languages will work on different processor architectures.
109. Which of the following is not true about software libraries?
A. They make software development more efficient through code reuse.
B. They are typically accessed through an application programming interface (API).
C. They almost never introduce vulnerabilities into programs that use them.
D. They are used in most major software development projects.
110. Kim is tasked with testing the security of an application but has no access to its
source code. Which of the following tests could she use in this scenario?
A. Dynamic application security testing
B. Static application security testing
C. Regression testing
D. Code review
111. Hanna is a security manager of a company that relies heavily on one specific
operating system. The operating system is used in the employee workstations and is
embedded within devices that support the automated production line software. She has uncovered a vulnerability in the operating system that could allow an attacker
to force applications to not release memory segments after execution. Which of the
following best describes the type of threat this vulnerability introduces?
A. Injection attacks
B. Memory corruption
C. Denial of service
D. Software locking
112. Which of the following access control mechanisms gives you the most granularity
in defining access control policies?
A. Attribute-based access control (ABAC)
B. Role-based access control (RBAC)
C. Mandatory access control (MAC)
D. Discretionary access control (DAC)
113. All of the following are weaknesses of Kerberos except which one?

A. Principals don’t trust each other.
B. Only the KDC can vouch for individuals’ identities and entitlements.
C. Secret keys are stored on the users’ workstations temporarily.
D. Susceptibility to password guessing and brute-force attacks.

114. A company needs to implement a CCTV system that will monitor a large area of
the facility. Which of the following is the correct lens combination for this?
A. A wide-angle lens and a small lens opening
B. A wide-angle lens and a large lens opening
C. A wide-angle lens and a large lens opening with a small focal length
D. A wide-angle lens and a large lens opening with a large focal length

115. What is the name of a water sprinkler system that keeps pipes empty and doesn’t
release water until a certain temperature is met and a “delay mechanism” is instituted?
A. Wet
B. Preaction
C. Delayed
D. Dry

116. There are different types of fire suppression systems. Which of the following answers
best describes the difference between a deluge system and a preaction system?
A. A deluge system provides a delaying mechanism that allows someone to

deactivate the system in case of a false alarm or if the fire can be extinguished
by other means. A preaction system provides similar functionality but has wide open sprinkler heads that allow a lot of water to be dispersed quickly.
B. A preaction system provides a delaying mechanism that allows someone to deactivate the system in case of a false alarm or if the fire can be extinguished
by other means. A deluge system has wide open sprinkler heads that allow a lot of water to be dispersed quickly.
C. A dry pipe system provides a delaying mechanism that allows someone to deactivate the system in case of a false alarm or if the fire can be extinguished
by other means. A deluge system has wide open sprinkler heads that allow a lot of water to be dispersed quickly.
D. A preaction system provides a delaying mechanism that allows someone to deactivate the system in case of a false alarm or if the fire can be extinguished
by other means. A deluge system provides similar functionality but has wide open sprinkler heads that allow a lot of water to be dispersed quickly.
117. Which of the following best describes why Crime Prevention Through Environmental Design (CPTED) would integrate benches, walkways, and bike paths into a site?
A. These features are designed to provide natural access control.
B. These features are designed to emphasize or extend the organization’s