Summary Developing software and applications for the cloud requires attention to security throughout the development process. That often starts with awareness and training, and then progresses through the rest of the software development lifecycle. Developers and security professionals need to be aware of common pitfalls as well as the most common cloud vulnerabilities, like those described by the OWASP Top 10 and SANS Top 25 lists. You'll also need to under stand the SDLC, common models, and what each phase of the SDLC involves. Along the way, you'll apply technologies like cryptography, sandboxing, and application containerization, as well as security components like web application firewalls and API gate ways. Threat modeling using models like STRIDE and ATASM, as well as secure coding, are part of the planning, design, and coding processes. Testing also plays a role; analysts need to be familiar with functional and nonfunctional testing, static and dynamic testing, and a variety of other parts of the testing and QA process. Finally, in addition to software development concepts, identity and access management is an important part of application architectures and implementation. While it plays a broad role in the cloud, the CCSP exam objectives pair it with application development, focusing on federated identity, identity providers, SSO, and MFA as well as secrets management. Since each of those is part of a secure design, familiarity with threats to IAM is also important for this domain. **Exam Essentials Explain cloud development basics, including common pitfalls and vulnerabilities to avoid.** Understand common pitfalls in application development like ensuring performance, scalability, portability, and interoperability. Be familiar with the OWASP Top 10 and SANS Top 25, including examples of the vulnerabilities from both lists. **Describe the software development lifecycle and how it can be applied.** Explain common software development models like Agile and Waterfall. Describe threat models including DREAD, STRIDE, PASTA, and ATASM. Be familiar with secure coding practices and standards like OWASP, ASVS, and SAFECode. **Apply testing methodologies to application software.** Describe functional and nonfunctional testing. Explain methodologies like full knowledge, zero knowledge, static and dynamic testing, and interactive application security testing. Understand QA processes and the role of QA in the SDLC. **Manage the software supply chain and use verified secure software.** Explain API security best practices and security concerns. Describe supply chain security practices and why supply chain security is important. Understand the importance of assessing vendors, particularly for open- source and third- party software components. **Explain common application security technologies and supplemental security controls.** Be familiar with design elements like when and where to use cryptography to protect data in motion and data at rest. Explain the use of and differences between sandboxing, application virtualization, microservices, and containers. Describe the role of orchestration in application environments. Be familiar with the uses for and roles of web application firewalls, database activity monitoring, XML firewalls, API gateways, and cloud application security brokers. **Understand IAM solutions as well as common threats to identity and access management.** Explain federated identity and the role of identity providers in federated environments. Describe the differences between SSO and MFA. Understand the basics of secrets management and why secrets need to be protected in application environments. Explain the differences between user, privileged, and service access. Be ready to explain threats like lost or exposed credentials, improper configurations, excessive permissions, and a lack of monitoring and alerting in the context of identity and access management for applications in the cloud.