6-1-基. 暗号化に関する知識

1. 科目の概要

OSS アプリケーションのセキュリティを確保するために必須の技術である暗号化について、公開鍵・秘密鍵暗号、認証などの各手法とその実装方法、および実用的な無線 LAN の暗号化やセキュアシェル(SSH)などの技術を解説する。

2. 習得ポイント

本科目の学習により習得することが期待されるポイントは以下の通り。

習得ポイント	説 明	シラバスの対応コマ
6-1-基-1. 暗号化の意義と効果、課題、注意点	暗号技術の基本概念と全体像を概説する。暗号化でどのような対応が可能か、暗号化処理の種類と利用時の課題、留意点などについて説明する。	1
6-1-基-2. 共通鍵暗号技術	暗号化方式のひとつである「共通鍵暗号方式」の基本概念と仕組み、特徴、利点と欠点、OSSにおける利用状況について説明する。また、米国で標準化されたAES (Advanced Encryption Standard)を紹介する。	2
6-1-基-3. 公開鍵暗号技術	暗号化方式のひとつである「公開鍵暗号方式」の基本概念と仕組みと特徴ついて説明する。	3
6-1-基-4. 電子署名技術	デジタル情報に対し電子署名を行うことによって認証性を担保する技術を説明する。基本概念や問題点を議論する。	9
6-1-基-5. 暗号技術によるデータ保護	OSやミドルウェア、アプリケーションで求められる暗号化処理とその実装を紹介する。またハードウェアレベルでの暗号化、ネットワークにおける暗号化にはどのようなものがあるか、その目的と特徴、実装方法を説明する。	4,10
6-1-基-6. 電子証明書とX.509	ネットワークにおける各ノードの正当性を証明する電子証明書について、それらの種類、その仕様、仕組み、電子証明書の役割と必要性およびX.509について述べる。	5
6-1-基-7. OSSにおける暗号技術利用方法	様々なOSS活用シーンにおける暗号化の必要性を示し、OSSによる暗号化処理の実装例を、OS、ミドルウェア、アプリケーションのレベルで分類して紹介する。	6
6-1-基-8. 無線LANに求められる暗号化の仕様、必要性、注意点	無線LANにおける暗号化の必要性について述べ、その仕様、特徴、利点と欠点などについて説明する。代表的な暗号化方式であるWEP (Wired Equivalent Privacy)と、さらに強化した暗号化方式のWPA (Wi-fi Protected Access) / WPA2 (Wi-fi Protected Access)などを紹介する。	8
6-1-基9. TLS(SSL)プロトコルと実例	TLS(Transport Layer Security)は暗号技術により通信を保護するトランスポート層に位置づけられるプロトコルである。httpsとして安全なHTTP通信を行うため、あるいはVPNの通信経路を保護するために使われるなど幅広く用いられている。	12
6-1-基-10. SSHプロトコルと実例	SSH(Secure Shell)は通信経路を暗号化し安全にリモートホストにログインするために作られたプロトコルである。ログイン認証はパスワードだけではなく電子署名でもログイン可能である。実装されたsshではログインだけではなく、安全なファイルの転送、リモートシェルの実行、ポートフォワーディングなどの機能を持つ。	11

【学習ガイダンスの使い方】

1. 「習得ポイント」により、当該科目で習得することが期待される概念・知識の全体像を把握する。
2. 「シラバス」、「IT 知識体系との対応関係」、「OSS モデルカリキュラム固有知識」をもとに、必要に応じて、従来の IT 教育プログラム等との相違を把握した上で、具体的な講義計画を考案する。
3. 習得ポイント毎の「学習の要点」と「解説」を参考にして、講義で使用する教材等を準備する。

3. IT 知識体系との対応関係

「6-1-基. 暗号化に関する知識」と IT 知識体系との対応関係は以下の通り。

科目名	1	2	3	4	5	6	7	8	9	10	11	12
6-1-基. 暗号化に関する知識	セキュリティ機能と暗号化の位置づけ	暗号化の方式・共通鍵暗号方式	暗号化の方式・公開鍵暗号方式	情報システムにおける暗号化適用の方式	電子証明書の仕組み	OSSの活用シーンと暗号化	OSSの暗号化ツール	無線LANの暗号化	認証と暗号化	IPsecによる暗号化通信	SSHによるトンネリング	SSLプロトコルの仕組み

<IT 知識体系上の関連部分>

分野		科目名	1	2	3	4	5	6	7	8	9	10	11	12	13
組織間連携事項と情報システム	1	IT-IAS. 情報保証と情報セキュリティ	IT-IAS1. 基礎的な問題	IT-IAS2. 情報セキュリティの仕組み(対策)	IT-IAS3. 運用上の問題	IT-IAS4. ポリシー	IT-IAS5. 攻撃	IT-IAS6. 情報セキュリティ分野	IT-IAS7. フォレンジック(情報証拠)	IT-IAS8. 情報の状態	IT-IAS9. 情報セキュリティサービス	IT-IAS10. 脅威分析モデル	IT-IAS11. 脆弱性		
	2	IT-SP. 社会的な観点とプロフェッショナルとしての課題	IT-SP1. プロフェッショナルとしてのコミュニケーション	IT-SP2. コンピュータの歴史	IT-SP3. コンピュータを取り巻く社会環境	IT-SP4. チームワーク	IT-SP5. 知的財産権	IT-SP6. コンピュータの法的問題	IT-SP7. 組織の中のIT	IT-SP8. プロフェッショナルとしての倫理的な問題と責任	IT-SP9. プライバシーと個人の自由				
応用技術	3	IT-IM. 情報管理	IT-IM1. 情報管理の概念と基礎	IT-IM2. データベース問合わせ言語	IT-IM3. データアーキテクチャ	IT-IM4. データモデリングとデータベース設計	IT-IM5. データと情報の管理	IT-IM6. データベースの応用分野							
	4	IT-WS. Webシステムとその技術	IT-WS1. Web技術	IT-WS2. 情報アーキテクチャ	IT-WS3. デジタルメディア	IT-WS4. Web開発	IT-WS5. 脆弱性	IT-WS6. ソーシャルソフトウェア							
ソフトウェアの方法と技術	5	IT-PF. プログラミング基礎	IT-PF1. 基本データ構造	IT-PF2. プログラミングの基本的構成要素	IT-PF3. オブジェクト指向プログラミング	IT-PF4. アルゴリズムと問題解決	IT-PF5. イベント駆動プログラミング	IT-PF6. 再帰							
	6	IT-IPT. 技術を統合するためのプログラミング	IT-IPT1. システム間通信	IT-IPT2. データ割り当てと交換	IT-IPT3. 結合的コーディング	IT-IPT4. スクリプティング手法	IT-IPT5. ソフトウェアセキュリティの実現	IT-IPT6. 種々の問題	IT-IPT7. プログラミング言語の概要						
	7	CE-SWE. ソフトウェア工学	CE-SWE0. 歴史と概要	CE-SWE1. ソフトウェアプロセス	CE-SWE2. ソフトウェアの要求と仕様	CE-SWE3. ソフトウェアの設計	CE-SWE4. ソフトウェアのテストと検証	CE-SWE5. ソフトウェアの保守	CE-SWE6. ソフトウェア開発・保守ツールと環境	CE-SWE7. ソフトウェアプロジェクト管理	CE-SWE8. 言語翻訳	CE-SWE9. ソフトウェアのフォールトトレランス	CE-SWE10. ソフトウェアの構成管理	CE-SWE11. ソフトエアの標準化	
	8	IT-SIA. システムインテグレーションとアーキテクチャ	IT-SIA1. 要求仕様	IT-SIA2. 調達/手配	IT-SIA3. インテグレーション	IT-SIA4. プロジェクト管理	IT-SIA5. テストと品質保証	IT-SIA6. 組織の特性	IT-SIA7. アーキテクチャ						
システム基盤	9	IT-NET. ネットワーク	IT-NET1. ネットワークの基礎	IT-NET2. ルーティングとスイッチング	IT-NET3. 物理層	IT-NET4. セキュリティ	IT-NET5. アプリケーション分野	IT-NET6. ネットワーク管理							
	10	CE-NWK. テレコミュニケーション	CE-NWK0. 歴史と概要	CE-NWK1. 通信ネットワークのアーキテクチャ	CE-NWK2. 通信ネットワークのプロトコル	CE-NWK3. LANとWAN	CE-NWK4. クライアントサーバコンピューティング	CE-NWK5. データのセキュリティと整合性	CE-NWK6. ワイヤレスコンピューティングとモバイルコンピューティング	CE-NWK7. データ通信	CE-NWK8. 組込み機器向けネットワーク	CE-NWK9. 通信技術とネットワーク概要	CE-NWK10. 性能評価	CE-NWK11. ネットワーク管理	CE-NWK12. 圧縮と伸張
			CE-NWK13. クラスタシステム	CE-NWK14. インターネットアプリケーション	CE-NWK15. 次世代インターネット	CE-NWK16. 放送									
	11	IT-PT. プラットフォーム技術	IT-PT1. オペレーティングシステム	IT-PT2. アーキテクチャと機構	IT-PT3. コンピュータインフラストラクチャ	IT-PT4. デプロイメントソフトウェア	IT-PT5. ファームウェア	IT-PT6. ハードウェア							
	12	CE-OPS. オペレーティングシステム	CE-OPS0. 歴史と概要	CE-OPS1. 並行性	CE-OPS2. スケジューリングとディスパッチ	CE-OPS3. メモリ管理	CE-OPS4. セキュリティと保護	CE-OPS5. ファイル管理	CE-OPS6. リアルタイムOS	CE-OPS7. OSの概要	CE-OPS8. 設計の原則	CE-OPS9. デバイス管理	CE-OPS10. システム性能評価		
コンピュータアーキテクチャ	13	CE-CAO. コンピュータのアーキテクチャと構成	CE-CA00. 歴史と概要	CE-CA01. コンピュータアーキテクチャの基礎	CE-CA02. メモリシステムの構成とアーキテクチャ	CE-CA03. インタフェースと通信	CE-CA04. デバイスサブシステム	CE-CA05. CPUアーキテクチャ	CE-CA06. 性能・コスト評価	CE-CA07. 分散・並列処理	CE-CA08. コンピュータによる計算	CE-CA09. 性能向上			
複数領域にまたがるもの	14	IT-ITF. IT基礎	IT-ITF1. ITの一般的なテーマ	IT-ITF2. 組織の問題	IT-ITF3. ITの歴史	IT-ITF4. IT分野(学科)とそれに関連のある分野(学科)	IT-ITF5. 応用領域	IT-ITF6. IT分野における数学と統計学の活用							
	15	CE-ESY. 組込みシステム	CE-ESY0. 歴史と概要	CE-ESY1. 低電力コンピューティング	CE-ESY2. 高信頼性システムの設計	CE-ESY3. 組込み用アーキテクチャ	CE-ESY4. 開発環境	CE-ESY5. ライフサイクル	CE-ESY6. 要件分析	CE-ESY7. 仕様定義	CE-ESY8. 構造設計	CE-ESY9. テスト	CE-ESY10. プロジェクト管理	CE-ESY11. 並行設計(ハードウェア、ソフトウェア)	CE-ESY12. 実装
			CE-ESY13. リアルタイムシステム設計	CE-ESY14. 組込みマイクロコントローラ	CE-ESY15. 組込みプログラム	CE-ESY16. 設計手法	CE-ESY17. ツールによるサポート	CE-ESY18. ネットワーク型組込みシステム	CE-ESY19. インタフェースシステムと混合信号システム	CE-ESY20. センサ技術	CE-ESY21. デバイスドライバ	CE-ESY22. メンテナンス	CE-ESY23. 専門システム	CE-ESY24. 信頼性とフォールトトレランス	

4. OSS モデルカリキュラム固有の知識

OSS モデルカリキュラム固有の知識として、OSS 特有のセキュリティに関する話題や、SSH プロトコルのオープンソース実装である OpenSSH などの暗号化技術に関連した OSS の実装に関する知識がある。

科目名	第1回	第2回	第3回	第4回	第5回	第6回	第7回	第8回	第9回	第10回	第11回	第12回
6-1-基 暗号化に関する知識	(1)オープンソースセキュリティの全体像	(1)共通鍵暗号方式の仕組み	(1)公開鍵暗号方式の仕組み	(1)ソフトウェア情報の暗号化	(1)電子証明書の種類	(1)オープンソースOS と暗号化	(1) ツールの機能例	(1)無線LAN 暗号化プロトコル WEP の仕様	(1)認証とは	(1)VPN の構成	(1)SSH とは	(1)SSL の概要
	(2)暗号化の意義と課題	(2)AES の概要	(2)インターネットでの公開鍵方式の重要性	(2)ハードウェアの暗号化	(2)電子証明書の仕様	(2)OSS における暗号化の実装	(2) OSSプロジェクトの例	(2)WPA の仕様	(2)メッセージダイジェストによる認証／改ざん防止機能	(2)Ipsec	(2)IKE による IPSec の設定	(2)SSL の仕様
				(3)通信路の暗号化	(3)証明書発行に関わる当事者と発行までの流れ				(3)メッセージ認証で確認されること	(3)セキュアな MPLS による IP-VPN		(3)SSL の安全性
					(4)CA 局による電子証明書発行、暗号化通信							(4)SSL 通信の構成

（網掛け部分は IT 知識体系で学習できる知識を示し、それ以外は OSS モデルカリキュラム固有の知識を示している）

スキル区分	OSS モデルカリキュラムの科目	レベル
セキュリティ分野	6-1　暗号化に関する知識	基本
習得ポイント	6-1-基-1. 暗号化の意義と効果、課題、注意点	
対応するコースウェア	第1回 セキュリティ機能と暗号化の位置づけ	

6-1-基-1. 暗号化の意義と効果、課題、注意点

暗号技術の基本概念と全体像を概説する。暗号化でどのような対応が可能か、暗号化処理の種類と利用時の課題、留意点などについて説明する。

【学習の要点】

* セキュリティに対する脅威としては、秘密が漏れる盗聴、情報が書き換えられる改竄、正しい送信者のふりをするなりすまし、後から当事者でないと宣言する否認、などが挙げられる。
* ここでは暗号技術として広く使われている、共通鍵暗号、公開鍵暗号、一方向ハッシュ関数、電子署名、擬似乱数生成器を取り上げる。
* 現在広く使われている暗号技術は計算量的安全性によってその安全性を担保されているが、コンピュータの計算能力の飛躍的向上と暗号解読技術の向上により徐々に安全性が低くなっていく。このことを暗号の危殆化という。

現在の NIST の推奨(2010年)	それ以前の状況
共通鍵暗号 AES 128bit 長以上	DES (56bit)、2-Key 3DES(112bit)
RSA 2048bit 以上	RSA 1024bit
DSA 2048bit 以上	DSA 1024bit
ECDSA 224bit 以上	ECDSA 192bit
SHA-2 (224bit / 256bit / 384bit / 512bit)	SHA-1 (160bit)

表 6-1-基-1.　NIST が現在推奨している暗号と鍵のサイズ(ハッシュ関数は出力サイズ)

【解説】

1) 暗号技術の意義と課題

暗号技術は「情報の秘匿」「情報の完全性」「情報の認証性」を提供する。情報セキュリティの脅威を具体的に考えてみる。オンラインショッピングでクレジットカード番号を入力しているとき、その通信が盗聴されクレジットカード番号が漏れる。メールで発注書を送ったが、途中で発注内容が改竄されてしまう。デジタル情報として契約書を作成した際に当事者以外がなりすましている、あるいは後から当事者でないと否認する。このような問題に対して暗号技術を用いることで解決を図ることが出来る。

＊ 計算量的安全性と暗号の危殆化問題

現在のコンピュータで使われている暗号は計算量的安全性の考え方に基づいて作られている。たとえば共通鍵暗号の場合、もし攻撃側が鍵の全件探索を行うことが可能であれば解読されてしまう。あるいは公開鍵暗号法の1つである RSA 法で、その公開鍵の因数分解が可能であれば解読されてしまう。安全な暗号であるためには解読に必要な計算量が現状で現実的には得られないレベルのものでなければならない。今日、飛躍的な CPU パワーの増大とネットワーク接続されたコンピュータの出現により、かつて安全であった暗号も安全ではなくなっている。また暗号アルゴリズムに対して分析を行い、より少ない計算量で解読する方法も日々進んでいる。このようにかつて安全であったものが徐々にと安全ではなくなる問題のことを暗号の危殆化問題と呼ぶ。

＊ 暗号技術の要素

暗号技術の要素としては、共通鍵暗号（対称鍵暗号）、公開鍵暗号、一方向ハッシュ関数、電子署名、擬似乱数生成器などさまざまな技術がある。

- 共通鍵暗号

  暗号化する時の鍵と復号する時の鍵が同じ方式である。米国政府標準の AES（FIPS PUB197)や Camellia(RFC3657, ISO/IEC 18033)などが広く使われている。

- 公開鍵暗号

  暗号化する時の鍵と復号する時の鍵が異なる。暗号化した鍵では復号できない。そのため暗号化する鍵を公開することが可能となる。暗号化する鍵を公開鍵(public key)、復号する鍵を秘密鍵(private key)と呼ぶ。TLS(SSL)では RSA を採用している。

- 一方向性ハッシュ関数

  任意の長さの入力を得て、一定の長さのハッシュ値の出力を行うとき、その出力から入力を見つけることが困難であるようなハッシュ関数のことである。暗号学的ハッシュ関数と呼ぶ場合もある。ハッシュ値はそのデータの特徴を示すので、指紋に例えてデジタルフィンガープリントと呼ぶ、あるいはメッセージを短くするのでメッセージダイジェストと呼ぶ場合もある。

- 電子署名

  データの改竄が行われていないかを見分けることができる技術である。TLS(SSL)では RSA と DSA を採用している。他にも楕円暗号の ECDSA が標準化され FIPS に採用されている。

- 擬似乱数生成器

  擬似的な乱数を生成する。共通鍵暗号や公開鍵暗号の鍵を生成するとき使われるので、セキュリティに重要な役割を果たす。良質な擬似乱数生成器が求めるには十分なエントロピーが必要である。

スキル区分	OSS モデルカリキュラムの科目	レベル
セキュリティ分野	6-1　暗号化に関する知識	基本
習得ポイント	基-6-1-2. 共通鍵暗号技術	
対応する コースウェア	第 2 回 暗号化の方式・共通鍵暗号方式	

基-6-1-2. 共通鍵暗号技術

暗号化方式のひとつである「共通鍵暗号方式」の基本概念と仕組み、特徴、利点と欠点、利用状況について説明する。また、米国で標準化された AES (Advanced Encryption Standard)を紹介する。

【学習の要点】

* 暗号化方式のひとつである共通鍵暗号では、1 つの鍵で情報を暗号化し、同じ鍵で復号する。
* 米国では政府標準の共通鍵暗号が選定されている。これまで標準とされていた DES (Data Encryption Standard, 1977)に代わって、新しい標準となる AES(Advanced Encryption Standard)が 2000 年に選定された。
* 共通鍵暗号の運用では安全な鍵の共有方法が問題となる。

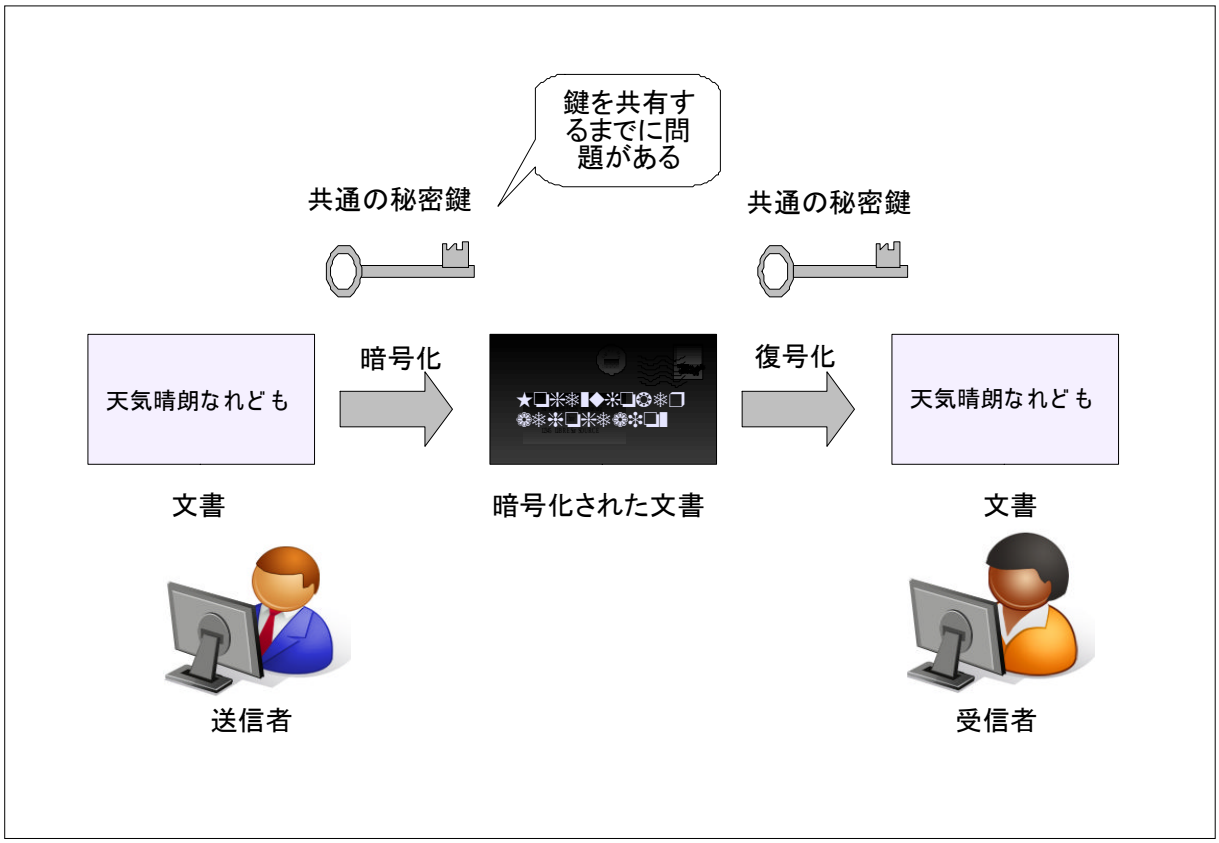

図 6-1-基-2.　共通鍵暗号方式の仕組み

【解説】

1) 共通鍵暗号

共通鍵暗号とは、暗号化と復号に同じ鍵を用いる暗号方式のことを言う。秘密鍵暗号、対称鍵暗号とも呼ばれることもある。

＊ 共通鍵暗号の特徴

- 現在の公開鍵暗号と比較して高速な暗号化・復号処理ができる。特にハードウェア化したときは高速である。
- 暗号化する側、復号する側とも同じ鍵を使うので、どのように事前に同じ鍵の受け渡しをするか、相手が増えると鍵を多数管理しなければならない、復号する鍵を自分以外も管理するなど鍵管理に関する問題がある。

2) 暗号モード

共通鍵暗号の実際の利用は暗号モードと組み合わせている。これは出力を入力にフィードバックする仕組みである。フィードバックしない場合、同じ入力は同じ出力になるためである。何もしないEBC モード、フィードバックをかける CBC モードなど色々なモードがある。

3) AES の概要

米国では政府標準の共通鍵暗号が選定されている。これまで標準とされていた DES (Data Encryption Standard, 1977)に代わって、新しい標準となる AES (Advanced Encryption Standard)が 2000 年に選定された。

＊ AES の安先生評価

AES は次期標準化暗号公募があり複数の候補の中で安全性の評価がなされ Rijndael(ラインデールもしくはリンデル)という暗号アルゴリズムが採用されるに至った。2010 年現在 AES に対する有効な攻撃は見つかっていない。

＊ 実装状況

現在では広く標準として利用されている。組み込み系 CPU ではハードウェア実装の AES が提供されているものもある。

4) 共通鍵暗号を選ぶ上での注意事項

安全な暗号アルゴリズムであるかどうか評価するには学術的にも難しく、その評価には多大なコストと時間がかかる。一方で、特に新しい共通鍵暗号の提案は多く、自称安全な暗号があちらこちらに溢れている。OSS にも評価が十分とはいえない暗号モジュールが組み込まれている場合がある。共通鍵暗号が選択できる場合、IETF、ISO、IEEE、FIPS といった標準規格として採用されている共通暗号をつかうことを推奨する。

過去に安全であった暗号が時間がたって安全ではなくなることを暗号の危殆化というが、WEP や SSL に使われている RC4(別名: ARCFOUR)や広く暗号化に使われていた DES は現在では安全な暗号ではない。このような暗号の利用は忌避すべきである。

スキル区分	OSS モデルカリキュラムの科目	レベル
セキュリティ分野	6-1　暗号化に関する知識	基本
習得ポイント	6-1-基-3. 公開鍵暗号技術	
対応する コースウェア	第 3 回 暗号化の方式・公開鍵暗号方式	

6-1-基-3. 公開鍵暗号技術

暗号化方式のひとつである「公開鍵暗号」の基本概念と仕組み、特徴について説明する。公開鍵暗号のネットワークにおける鍵配布・鍵管理の利点を説明する。また暗号の危殆化の考え方や安全性や使われ方を示す。

【学習の要点】

＊ 公開鍵暗号では、公開鍵で情報を暗号化し、秘密鍵で復号する。

＊ 公開鍵暗号を使う場合、復号のための鍵（秘匿が必要な鍵）を他者と共有する必要がなくなり、共通鍵暗号を利用する際に起きた鍵管理の問題を回避できる。

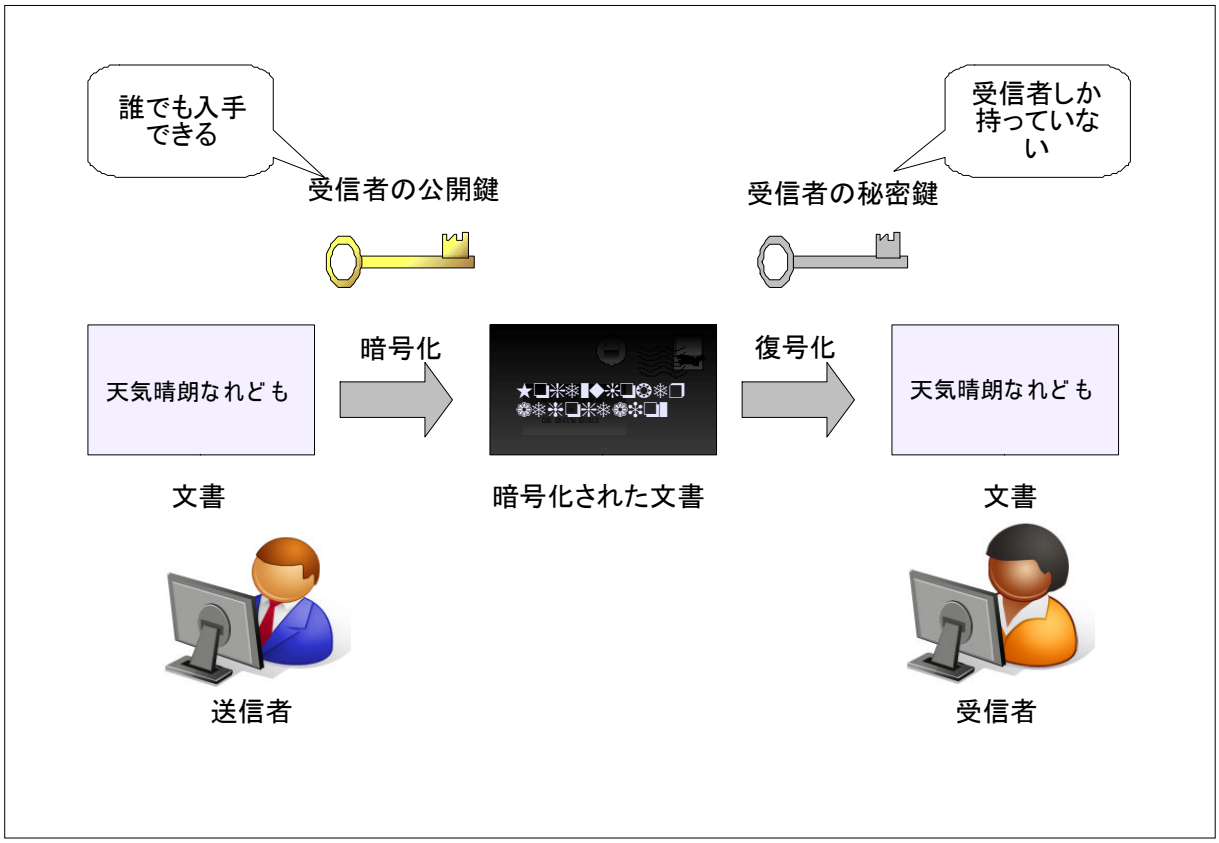

図 6-1-基-3.　公開鍵暗号の仕組み

【解説】

1) 公開鍵暗号の仕組み

公開鍵暗号では、公開鍵で情報を暗号化し、秘密鍵で復号する。

＊ 公開鍵暗号の特徴

- 相手にデータを暗号化するときに公開鍵を送ればよく、また公開鍵を第三者に入手されても秘密鍵は自分の手元のみにあるので暗号化したデータの秘匿性に問題がない。
- 複数の相手とやり取りするときでも同じ公開鍵を配布することができる、自分は秘密鍵1つを保持していれば良い。
- 現在広く使われている公開鍵暗号の処理には数学的計算を行うため共通鍵暗号の処理よりも計算コストがかかる。

＊ 公開鍵暗号の利用シナリオ

以下の手順でメッセージが送信される。ここでは Alice を送信者と Bob を受信者とする。

- (Bob)公開鍵と秘密鍵の鍵ペアを作成→(Bob)公開鍵を暗号化に利用してもらうためにアリスに送付→(Alice)ボブの公開鍵を使って、メッセージを暗号化→(Alice)暗号文をボブに送付→(Bob)秘密鍵で暗号文を復号

2) 公開鍵暗号の安全性

公開鍵暗号で TLS(SSL)、SSH、OpenPGP などに使われている公開鍵暗号 RSA を例に取って説明する。RSA 暗号の安全性は2つの大きな素数からなる合成数を素因数分解するのにかかる計算コストに依存する。512bit は 1999 年、640bit は 2005 年、768bit は 2009 年に素因数分解が成功している。2010 年現在では RSA の鍵のサイズは 2048bit 以上が推奨される。このように計算能力の向上によりより長い公開鍵が必要となっている。

3) 公開鍵暗号の使われ方

TLS(SSL)、OpenPGP、SSH などのプロトコルで公開鍵暗号は使われている。TLS や SSH ではサーバ・クライアント間で通信を暗号化するときにつかうセッション鍵(共通鍵暗号で利用)を生成するのに必要な秘密情報を交換する。OpenPGP(コマンド gnupg)ではデータを暗号化するのは共通鍵暗号で行い、その際に使われたセッション鍵を公開鍵暗号で暗号化し、暗号化したデータと暗号化したセッション鍵を送る。

4) 危殆化と公開鍵暗号

公開鍵暗号の場合、危殆化は鍵の長さをさらに伸ばすことで解決が図れる。ただし、その場合、暗号化あるいは復号、もしくは両方の計算量が増える。RSA や DSA といった暗号は鍵が長くなるにつけ計算量の伸びが大きいという問題がある。一方で鍵の長さが小さい楕円曲線暗号系の暗号は安全性に見合う鍵の長さを増やしても、計算量の伸びが小さい。将来的には現在の RSA や DSA といったものは楕円曲線暗号系のアルゴリズムを持つ暗号に変化していく可能性が大きい。

スキル区分	OSS モデルカリキュラムの科目	レベル
セキュリティ分野	6-1　暗号化に関する知識	基本
習得ポイント	6-1-基-4. 電子署名技術	
対応する コースウェア	第 9 回　認証と暗号化	

6-1-基-4. 電子署名技術

デジタル情報に対し電子署名を行うことによって認証性を担保する技術を説明する。電子署名は完全性、認証性を提供する。電子署名の内部機能は大きく分けて一方向性ハッシュ関数によりハッシュ値を取る機能、電子署名アルゴリズムでハッシュ値を保護する機能から成り立っている。

【学習の要点】

* 電子署名はデータに対して、改竄・偽造の検出（完全性）、本人確認（認証性）を行うために用いられる。
* 一般的な電子署名の使われ方は、デジタル情報に対して一方向性ハッシュ関数でハッシュ値を取り、その値を電子署名アルゴリズムで保護する。
* その電子署名による本人認証などでは電子証明書が必要である。

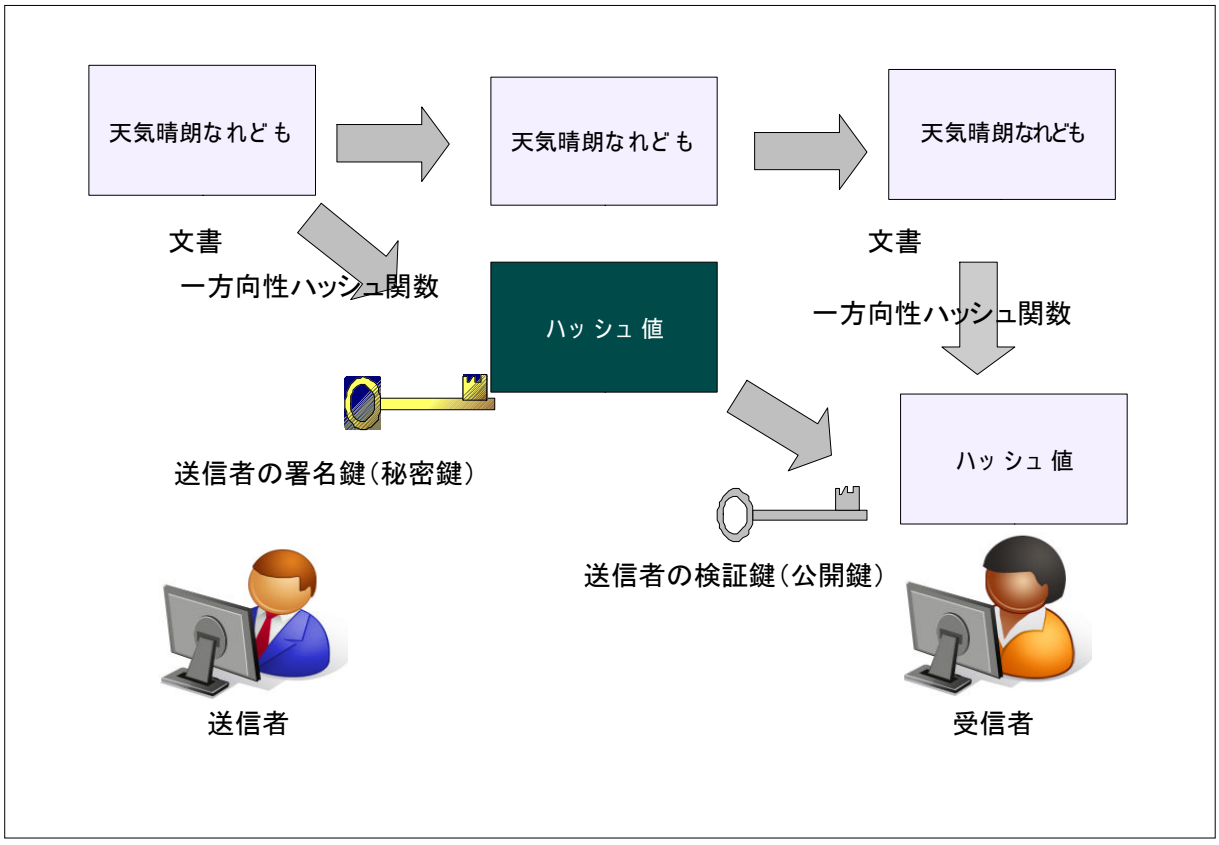

図 6-1-基-4.　電子署名の仕組み

【解説】

1) 電子署名技術

デジタル情報は、そのままでは内容を書き換えても書き換えたことがわからない。電子署名技術はデジタル情報が改竄されたか否かを検出する技術である。利用するためにはまず、署名を行う署名鍵(秘密鍵)、検証を行う検証鍵(公開鍵)の対を作成し、相手に検証鍵を渡しておく。

* 署名
  - データを一方向性ハッシュ関数で処理し、ハッシュ値を求める。
  - ハッシュ値を電子署名アルゴリズムで処理する。その際、署名鍵を使う。ハッシュ値は保護された値になる。
  - データと保護されたハッシュ値を一緒に用意しておく。
* 検証
  - データと保護されたハッシュ値の両方を入手する。
  - データを一方向性ハッシュ関数で処理し、ハッシュ値を求める。
  - 保護されているハッシュ値を電子署名アルゴリズムで戻す。その際、検証鍵を使う。
  - 両方のハッシュ値を比較し同じ値であればデータは改竄されていないことがわかる。

署名鍵と検証鍵は一対であり、この組み合わせ以外では検証が成功しない。電子署名が行われているデジタル情報を検証し、その検証が正しい場合、その署名者は正しい署名鍵を持っていることになる。ただし、この時点では相手が正しい署名鍵を持っていることは証明できるが、それの所有者の存在証明とはならない。そのため別途、X.509 のような公開鍵証明書や OpenPGP の Web of Trust のような方式を使い署名鍵所有者が正しい所有者であるか確認する必要がある。

2) RSA 及び DSA/ECDSA

電子署名アルゴリズムは、公開鍵暗号と同じアプローチで署名者が秘密にしておき署名を行うときに使う鍵と、任意の署名検証を行う者に公開し検証を行う鍵が用意され利用される。これは一対になっており他の組み合わせでは検証が成功しない。RSA では秘密鍵で復号する手順が署名を行うことに相当し、その署名を公開鍵で暗号化する手順を行うことが検証に相当する。暗号化→復号のプロセスの反対を行うことになる。米国国立標準技術研究所 NIST によって作られた、離散対数問題に基づく電子署名アルゴリズム DSA 及び楕円離散対数問題に基づく電子署名アルゴリズム ECDSA は電子署名のみ行え、暗号には利用できない。

3) SHA256 (SHA-2)

一方向性ハッシュ関数のハッシュ値は、よくデータの「指紋」といわれるが、違うデータから同じハッシュ値が得られる確率は無視できる。ハッシュ値が同じデータは内容が同じ、違えばデータは異なっている。現在 SHA256(SHA-2 ファミリのハッシュ関数で出力値が 256bit)が主流である。他にも SHA224/384/512 が使われる。SHA-1 は 160bit のため今日ではその寿命が尽きようとしている。以前最も使われていた MD5 は既に安全ではない。過去のシステムとの互換性のために MD5 が利用できる場合もあるが利用は推奨しない。

スキル区分	OSS モデルカリキュラムの科目	レベル
セキュリティ分野	6-1　暗号化に関する知識	基本
習得ポイント	6-1-基-5 暗号技術によるデータ保護	
対応する コースウェア	第 4 回 情報システムにおける暗号化適用の方式 第 10 回 IPSec による暗号化通信	

6-1-基-5 暗号技術によるデータ保護

OS やミドルウェア、アプリケーションで求められる暗号化処理とその実装を紹介する。またハードウェアレベルでの暗号化、ネットワークにおける暗号化にはどのようなものがあるか、その目的と特徴、実装方法を説明する。

【学習の要点】

* 業務で使われるソフトウェアは機密情報やプライバシーにかかわる情報を扱うことが多い。情報漏洩を防ぎ、信頼性を維持するために OS、データベース、アプリケーションレベルでの暗号化が行われる。
* 主記憶、ハードディスク、外部記憶媒体といったハードウェアレベルでの暗号化も、情報漏洩の危機管理などの観点から行われる。
* ネットワークレベルの暗号化も通信の安全性・機密性を高めるために行われる。ネットワークレベルの暗号化には IPsec, SSL/TLS などの選択肢がある。

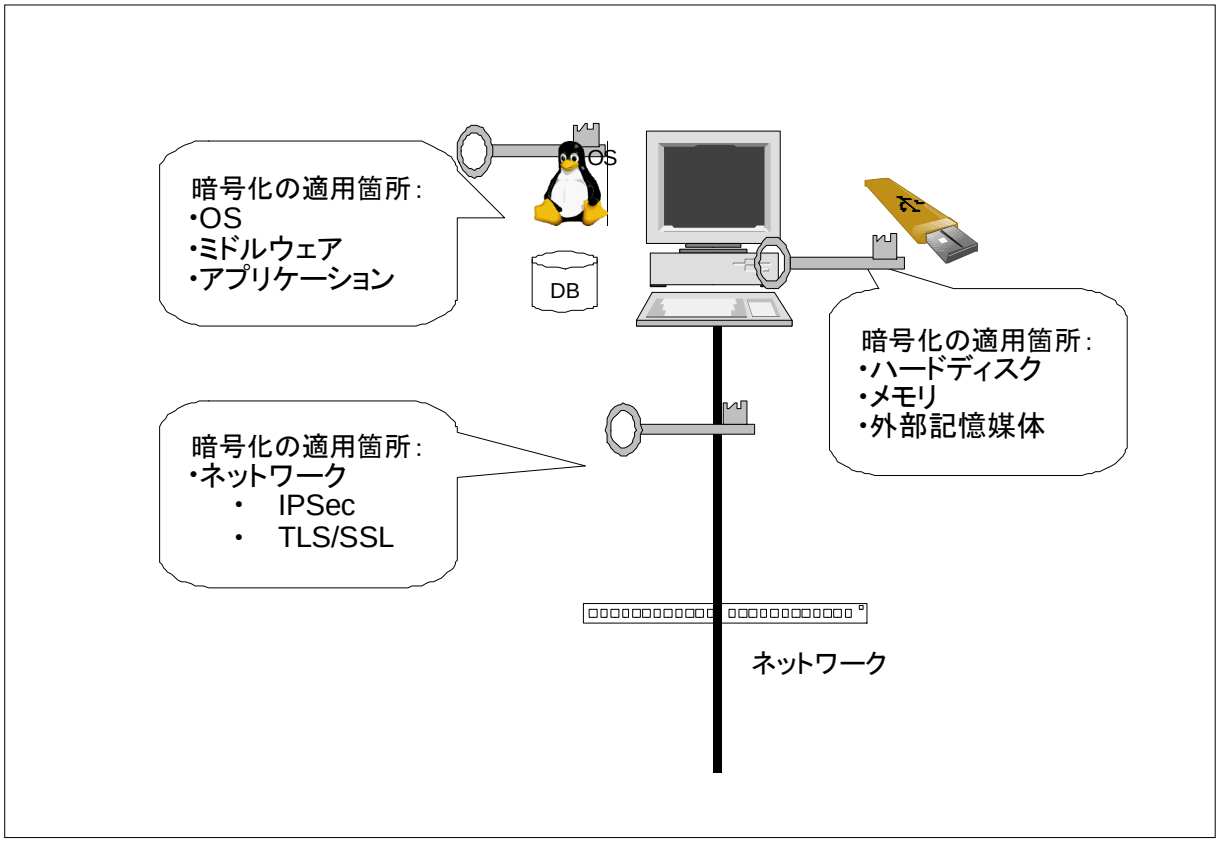

図 6-1-基-5.　ソフトウェア、ハードウェア、ネットワークに対する暗号化の適用

【解説】

1) ソフトウェア情報の暗号化

業務で使われるソフトウェアは機密情報やプライバシーにかかわる情報を扱うことが多い。情報漏洩を防ぎ、信頼性を維持するためにソフトウェア情報の暗号化は重要である。OS、データベース、および機密データを扱うアプリケーションのそれぞれにデータを暗号化して処理する枠組みが用意されている。

2) ハードウェアの暗号化

ソフトウェアのみで改竄されないことを担保するのは原理的にも実際的にも困難なため、主記憶、ハードディスク、外部記憶媒体といったハードウェアレベルでの暗号化も行われる。

＊ ハードディスク、外部記憶媒体の暗号化

ハードディスクや外部記憶媒体に送られてきたデータをすべて暗号化して書き込む技術も、商品化されている。暗号化処理は、ハードウェア回路を使って実行している。

＊ ハードウェア暗号化の利点／欠点

ハードウェア回路を使った暗号の場合、暗号化処理は高速、高信頼に実行できる。その反面、暗号化回路が必要となり比較的高価になる。

3) 通信路の暗号化

ネットワークレベルの暗号化も通信の安全性・機密性を高めるために行われる。ネットワークレベルの暗号化には IPsec, SSL/TLS などの選択肢がある。

＊ ネットワーク暗号化の利点／欠点

通信路の機密性は保たれる。しかし、処理コストがかかるため、ネットワーク性能のボトルネックになることもある。

スキル区分	OSS モデルカリキュラムの科目	レベル
セキュリティ分野	6-1　暗号化に関する知識	基本
習得ポイント	6-1-基-6 電子証明書と X.509	
対応する コースウェア	第 5 回 電子証明書の仕組み	

6-1-基-6 電子証明書と X.509

ネットワークにおける各ノードの正当性を証明する電子証明書について、それらの種類、その仕様、仕組み、電子証明書の役割と必要性について述べる。

【学習の要点】

* 電子証明書は暗号通信の際に使われる公開鍵が正しいものであることを証明するものである。
* 証明書の標準規格としては X.509 が存在しており、多くのアプリケーションでサポートされている。
* 証明書発行の際には認証局を利用することになる。この証明書発行までの一連の流れを理解する。

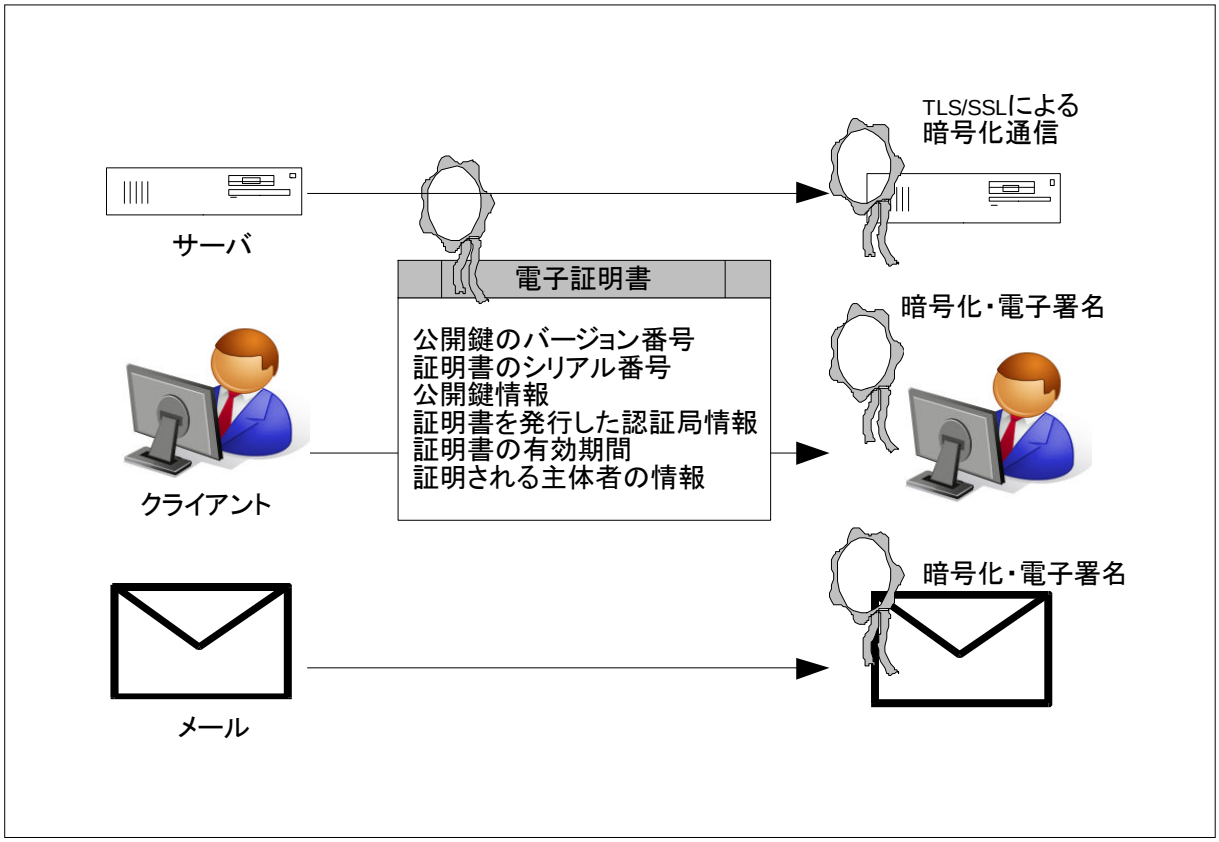

図 6-1-基-6.　電子証明書

【解説】

1) 電子証明書の仕組み

電子証明書は暗号通信の際に使われる公開鍵が正しい所有者のものであることを証明する。

＊ 電子証明書の種類

電子証明書の種類には、サーバ証明(VPN, Web)、クライアント証明、メール証明、ソフトウェア証明などがある。

＊ 電子証明書の仕様

- 証明書の標準規格としては X.509 が存在しており、多くのアプリケーションでサポートされている。
- X.509 は公開鍵のバージョン番号、証明書のシリアル番号、公開鍵情報、証明書を発行した認証局情報、証明書の有効期間、証明される主体者の情報、拡張領域といった項目で構成される。

2) 証明書発行に関わる当事者と発行までの流れ

証明書発行の際には認証局を利用することになる。公開鍵基盤(PKI: Public-Key Infrastructure)は公開鍵を運用するために定められた企画や仕様の総称である。PKI における証明書発行までの一連の流れは以下の通り。

＊ 証明書発行に関わる当事者

- 認証局
  証明書を発行する人
- 利用者
  PKI を利用する人
- リポジトリ
  証明書を保管しているデータベース

＊ 発行までの手順。

利用者が公開鍵を認証局に登録し、認証局は利用者の公開鍵に認証局のデジタル署名を付けたものを証明書としてリポジトリに保存する。公開鍵を利用する、登録者とは別の利用者はリポジトリから証明書をダウンロードする。

スキル区分	OSS モデルカリキュラムの科目	レベル
セキュリティ分野	6-1　暗号化に関する知識	基本
習得ポイント	6-1-基-7 OSS における暗号技術利用例	
対応する コースウェア	第 6 回　OSS　の活用シーンと暗号化	

6-1-基-7 OSS における暗号技術利用例

様々な OSS 活用シーンにおける暗号化の必要性を示し、OSS による暗号化処理の実装例を、OS、ミドルウェア、アプリケーションのレベルで分類して紹介する。

【学習の要点】

＊ OSS の活用されている多くのサーバソフトウェアにおいて暗号化処理が実装されている。

＊ OSS の OS、ミドルウェア、アプリケーションにおける暗号化処理の実際をデータベースソフトウェア、ネットワークアプリケーションなどを通して理解する。

図 6-1-基-7.　OSS の暗号化モジュール、ツール

【解説】

1) オープンソース OS と暗号化

オープンソース OS の活用されているサーバおよびクライアントにおいても暗号化機能は様々なシーンで活用される。例えば Linux ではカーネルにも暗号化モジュールが組み込まれている。暗号化ファイルシステムといった機能も提供される。

＊ OS の活用シーンと暗号化

- サーバとして
  イントラネットサーバ・インターネットサーバとして稼働する際に、SSL/TLS は標準的に利用されている。また、SSH によるアクセスを受け付ける際にも用いられる。
- クライアントとして
  SSL/TLS、SSH でサーバにアクセスする際、およびディスクデータの暗号化などが行われる。

＊ 暗号化の有効性評価

有効性評価には、いくつかの確認すべき項目がある。

- 暗号技術評価プロジェクト CRYPTREC（Cryptography Research and Evaluation Committees）などの機関で推奨する暗号を利用しているかどうか。
- TLS などの暗号化モジュールはセキュリティホールに対してパッチが当てられた最新版を利用しているかどうか。
- 鍵の管理など、セキュリティポリシーが制定・遵守されているかどうか。

2) OSS における暗号化の実装

OS のほか、ミドルウェア、アプリケーションなどでも暗号化の実装を活用するシーンがある。

＊ カーネル
　暗号化ファイルシステム、IPSec

＊ ミドルウェア
データベースソフトウェア、運用管理ソフトウェア、信頼性・性能向上ツール、

＊ アプリケーション
ネットワークアプリケーション、汎用業務アプリケーション

スキル区分	OSS モデルカリキュラムの科目	レベル
セキュリティ分野	6-1　暗号化に関する知識	基本
習得ポイント	6-1-基-8 無線 LAN に求められる暗号化の仕様、必要性、課題	
対応する コースウェア	第 8 回 無線 LAN の暗号化	

6-1-基-8 無線 LAN に求められる暗号化の仕様、必要性、課題

無線 LAN における暗号化の必要性について述べ、その仕様、特徴、利点と欠点などについて説明する。代表的な暗号化方式である WEP (Wired Equivalent Privacy)と、さらに強化した暗号化方式の WPA (Wi-fi Protected Access) / WPA2 (Wi-fi Protected Access)などを紹介する。

【学習の要点】

* 無線 LAN はオフィスや家庭などでその利便性から広範に利用されるが、安易な設定ではデータ盗聴のリスクがある。
* 過去に広く使われていた WEP では、共通鍵暗号 RC4 の危殆化問題、あるいはプロトコル自体に脆弱性があり今日では安全性を保てない。
* WEP の問題点を補うためにセキュリティ強化された WPA では、TKIP(Temporal Key Integrity Protocol)という暗号化方式を取り入れたが、解読がより短時間で行える可能性が出てきているため、今後の利用は考慮すべきである。
* 今日では WPA で共通鍵暗号 AES を使うか、あるいは AES ベースの CCMP を採用している WPA2 を使うことが求められる。

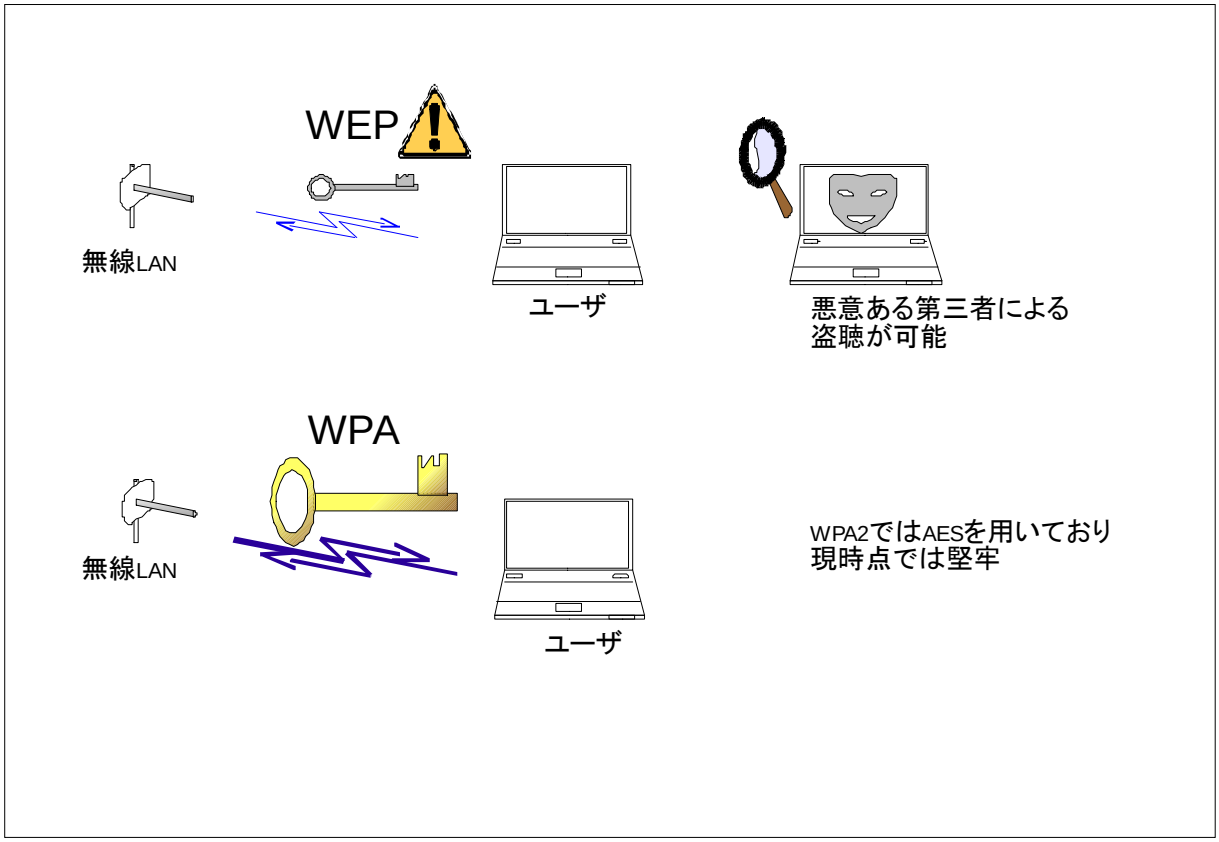

図 6-1-基-8.　無線 LAN の暗号化

【解説】

1）無線 LAN 暗号化プロトコル WEP の仕様

＊ 無線 LAN 暗号化のリスク

- 無線 LAN のセキュリティを確保するために標準的に用いられている暗号化プロトコルである WEP には以前から脆弱性が指摘されており、データの盗聴や不正利用といったリスクがある。

＊ WEP プロトコルの暗号化手順

- WEP では、データ・パケットを共通鍵で暗号化する際に RC4 ストリーム暗号を使う。

＊ WEP のリスクとその対応

- RC4 暗号が既に解読されていることと、WEP の動作手順自体にも脆弱性が認められている。
- WPA や WPA2 などより高度なセキュリティ・プロトコルを使い，定期的にパスワードを変えることが望ましい。

2）WPA の仕様

＊ WEP の問題点を補うために構築された WPA では、TKIP（Temporal Key Integrity Protocol）という暗号化方式を用いる。

＊ TKIP により鍵長の拡張のほか、一定時間ごとに暗号鍵を更新することで、暗号の強度を高めている。

＊ しかし、内部で用いている暗号方式が RC4 である場合、解読が可能であることが指摘されている。

＊ そのため、WPA2 では、AES（Advanced Encryption Standard）を採用し安全性を強化している。

＊ 2010 年時点では WPA2 は安全であるが今後、暗号解読技術の向上によって解読になったり、あるいはプロトコルそのものに脆弱性が見つかるなどの可能性も否定はできない。セキュリティ関しては常に最新の情報をチェックしておかなければならない。

スキル区分	OSS モデルカリキュラムの科目	レベル
セキュリティ分野	6-1　暗号化に関する知識	基本
習得ポイント	6-1-基-9 TLS(SSL)プロトコルと実例	
対応する コースウェア	第 12 回 SSL プロトコルの仕組み	

6-1-基-9 TLS(SSL)プロトコルと実例

TLS(Transport Layer Security)は暗号技術により通信を保護するトランスポート層に位置づけられるプロトコルである。https として安全な HTTP 通信を行うため、あるいは VPN の通信経路を保護するために使われるなど幅広く用いられている。

【学習の要点】

* TLS は IETF での作業部会の名前からきており、その前身となった SSL(Secure Sockets Layer)と呼ばれることもある。
* TLS は TCP の上に位置するがレイヤ的にはトランスポート層の位置づけとしている。
* X.509 電子証明書を用いた接続先認証を行うこともできる。
* HTTP 通信の暗号化として HTTPS に用いられるのが身近な例であるが、特定のアプリケーションに結びついて作られているわけではない。

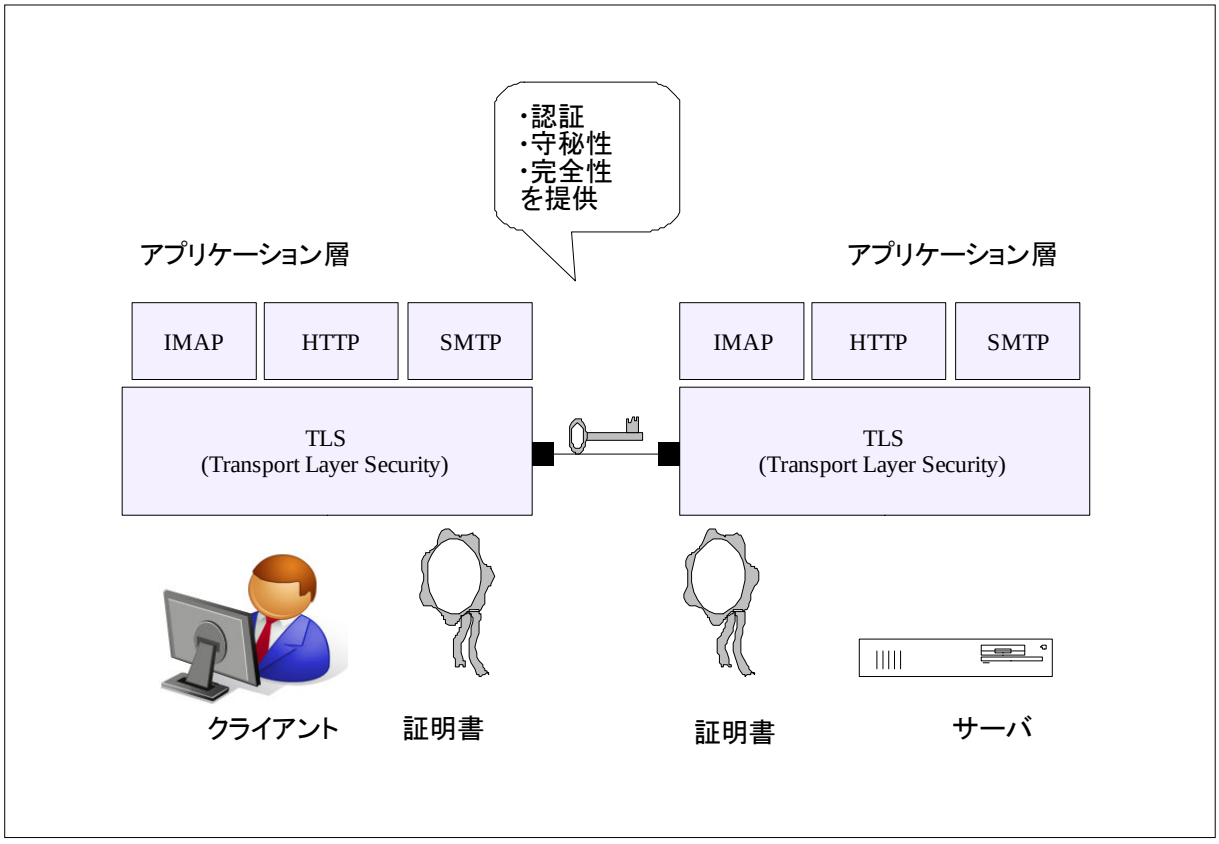

図 6-1-基-9.　サーバの暗号化(TLS)

【解説】

1) TLS の概要

＊ TLS は暗号通信の方法として最も広く使われているプロトコルである。

＊ TLS は SSL3.0 を元に、IETF によってつくられたプロトコルである。

＊ HTTP による Web サーバの通信を暗号化する時などに利用されているが、汎用のトランスポート層での保護を提供している。

＊ OpenSSL は TLS の OSS デファクト実装である。

＊ X.509 公開鍵証明書を使いサーバやクライアントの正当性を確認する。

2) TLS の仕様

TLS プロトコルは、「TLS レコードプロトコル」と「TLS ハンドシェイクプロトコル」という 2 つのプロトコルからなる。

＊ TLS レコードプロトコル

- TLS レコードプロトコルは、TLS ハンドシェイクプロトコルの下にあり、共通暗号を用いて暗号化されたメッセージ通信を行う部分である。

＊ TLS ハンドシェイクプロトコル

- TLS ハンドシェイクプロトコルは、暗号方式の決定、暗号方法の変更、エラー発生通知、データ転送という機能を持つ 4 つのプロトコルからなる。

3) TLS の安全性

＊ TLS は暗号通信のフレームワークを提供している。

＊ この枠組みでは、TLS で使われる共通暗号、公開鍵暗号、電子署名、一方向性ハッシュ関数などを部品のように切り替えることができる。

＊ TLS 1.2 で実装が必須とされているアルゴリズムは次の通りである。

- 暗号化 ： AES
- 認証 ： RSA
- メッセージダイジェスト関数 ： SHA-1

＊ セキュリティ強度

- TLS の暗号アルゴリズムもバージョンが上がるにつけアルゴリズムや鍵の長さが強化されている。
- 2010 年時点においては AES128、RSA2048、SHA256 の組みあわせ、もしくはそれ以上の暗号強度を推奨する。
- 利用者としては、X.509 公開鍵証明書の発行元が信頼できるところであるかなど留意する必要がある。

スキル区分	OSS モデルカリキュラムの科目	レベル
セキュリティ分野	6-1　暗号化に関する知識	基本
習得ポイント	6-1-基-10. SSH プロトコルと実例	
対応する コースウェア	第 11 回　SSH によるトンネリング	

6-1-基-10. SSH プロトコルと実例

SSH(Secure Shell)は通信経路を暗号化し安全にリモートホストにログインするために作られたプロトコルである。ログイン認証はパスワードだけではなく電子署名でもログイン可能である。実装された ssh ではログインだけではなく、安全なファイルの転送、リモートシェルの実行、ポートフォワーディングなどの機能を持つ。

【学習の要点】

* SSH（Secure Shell）は安全にリモートコンピュータにログインするために通信経路を暗号で保護するプロトコルである。
* ログインにはパスワードだけではなく事前に登録しておいた検証鍵（公開鍵）をリモートホストに登録しておくと電子署名によるユーザ認証でログインできる。
* OpenSSH など実際の実装では、リモートログインのほかにファイルの転送(sftp/scp)やリモートシェルの実行、ポートフォワーディングの機能も提供する。

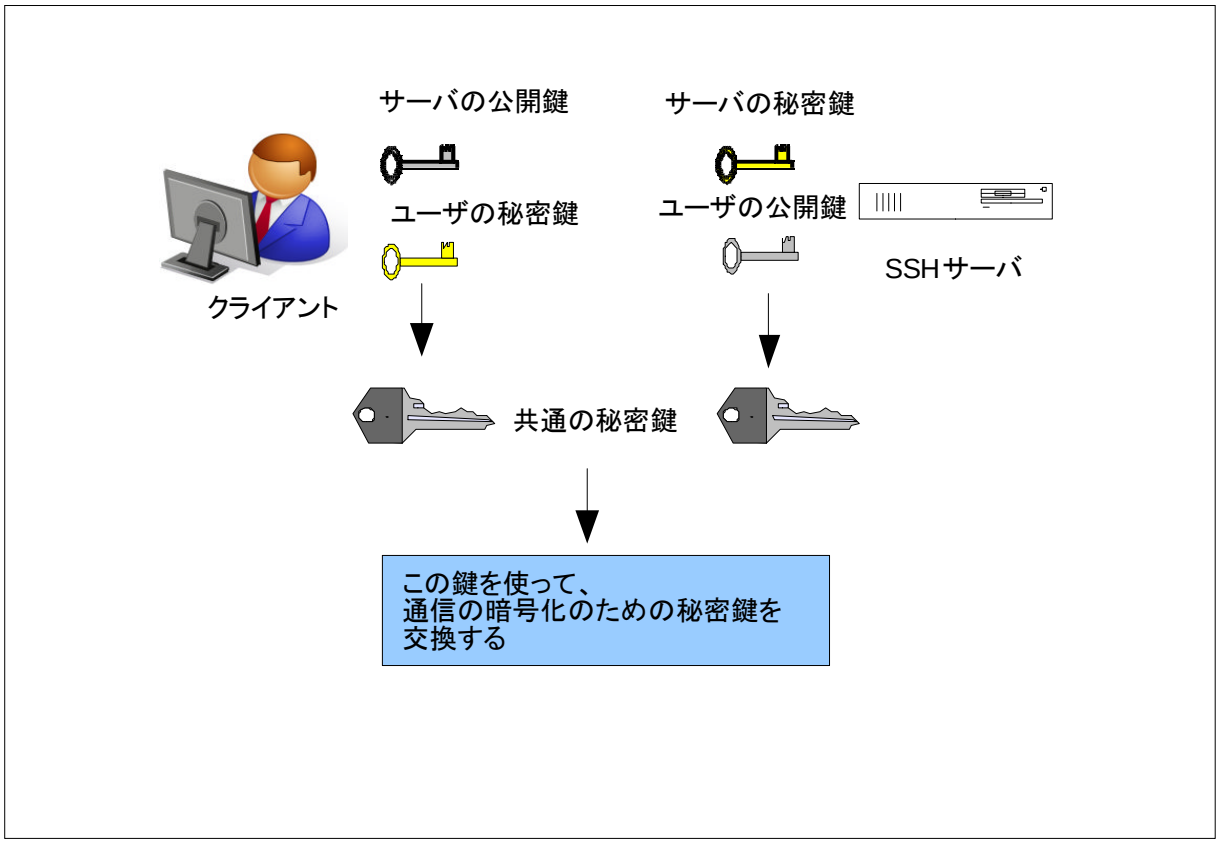

図 6-1-基-10.　SSH における暗号化された通信に必要な処理手順

【解説】

1) SSH とは

SSH は通信経路の暗号化や認証を強化したネットワーク越しの処理を可能とする、トランスポート層のプロトコルである。

* 機能と特徴
  - 通信路を暗号化するので安全にリモートログインができる。
  - ログインにパスワードを使わず電子署名を使ってログインすることが可能なのでパスワード攻撃からアカウントを守ることができる。
  - リモートホストと暗号化した通信路を確保し、ファイルを転送することができる。リモートファイルをコピーする SCP、FTP を代替する SFTP などがある。
  - リモートホストとのあいだでポート転送をすることができるので一種の VPN を利用できる。
* OpenSSH
  - SSH プロトコルを実装するデファクトで、OpenBSD プロジェクトにより開発・メンテナンスされている。
  - SSH のクライアントアプリケーションは各種プラットフォームに多数存在する。

2) 安全なログイン

電子署名を使ってログインする方法を使えば、ログインパスワードが漏れてあるいは類推され侵入されるというケースはなくなる。

* ssh-keygen を使いユーザの公開鍵と秘密鍵を生成する。
* ローカルホストにある公開鍵(~/.ssh/identity.pub)をリモートホストの公開鍵登録ファイル(~/.ssh/authorized_keys)に加える。
* 次回から SSH でログイン時にはローカルホストの秘密鍵を読み込むためのパスワードが聞かれる。
* リモートホストへのログイン認証はその秘密鍵と登録済みの公開鍵を使っての電子署名で確認する。

3) ポートフォワーディング

ローカルホストのポートをリモートのポートにフォワードすることが可能である。これにより安全な通信路を経由してネットワークアプリケーションが使えるようになる。

* リモートサーバ remotehost.domain 上にある pop3(110/tcp)サーバをアクセスするためにポートフォワードを行う。ローカルホストのアプリケーションはローカルホスト上の 110/tcp をアクセスする。

  $ sudo ssh –L 110:localhost:110 –l user remotehost.domain
* ローカルホストでアクセスするポートを 8008/tcp とし、リモートサーバ remotehost.domain をプロキシとして www.example.com の HTTP サーバ(80/tcp)にアクセスする場合は次のようになる。

  $ ssh –L 8008:www.example.com:80 remotehost.domain