Primitive凡例 S:Sensor、A:Aggregator、C:Communication Channel、E:e-Utility、D:Decision trigger

No.		Primitive S	A	C	E	D	大項目	小項目	本項目の目的	開発する際に確認する項目	利用する際に確認する項目	回答欄	回答の補足
I	1	○	○		○	○	ユーザ管理	アカウントロックアウトメカニズム	第三者が端末を不正に操作できないようにする	連続した規定回数以上のログイン失敗や多重ログインなどの痕跡を確認したら、アカウントをロックし、ログインが不可になる機能を持たせる	アカウントロックに関する設定可能な内容を確認し、自身で設定したとおりにアカウントがロックされるか確認する		
	2		○		○	○		一定期間利用されていないアカウントの強制失効オプション	一定期間利用されていないアカウントからのログインをできないようにする	設定した有効期限を超過したアカウントをロックする機能を持たせる	有効期限後にアカウントが失効することを確認する		
	3		○		○	○		パスワード強度の担保機能	ブルートフォース、辞書攻撃などにより不正にログインされないようにする	複数の文字種の利用や一定文字数以上などの条件を満たしたパスワードのみ登録できる機能を持たせる	条件に満たないパスワードの登録ができないことを確認する		
	4		○		○	○		パスワードセキュリティオプション（二要素認証など）	第三者がシステムにログインすることを困難にする	パスワードセキュリティオプションを利用できるようにする （例：二要素認証など）	パスワードセキュリティオプションが利用できることを確認する		
	5		○		○	○		サービスやプロセスを起動するアカウントの権限管理	アカウント毎にサービスやプロセスを動かす権限を限定してインシデント発生時の影響範囲をサービスやプロセスの範囲内におさえる	サービスやプロセスの起動にスーパーユーザを求めない作りにする	サービスやプロセス専用のユーザで動作することを確認する		
	6		○		○	○		共有ユーザアカウント	用途に応じて適切な権限を付与できるようにする	アカウント管理機能を持たせる	共有するアカウントが適切な権限と共有範囲で利用されていることを確認する		
	7		○		○	○		管理ユーザへの適切な権限付与	管理ユーザが必要な権限を使えるようにする	権限を管理する機能を持たせる	管理者ユーザが適切な権限を付与されているか確認する		
	8		○		○	○		一般ユーザへの権限付与機能	ユーザに必要な権限のみを使えるようにする	ユーザに権限を付与できる機能を持たせる	ユーザに権限が付与でき、権限に応じた利用ができることを確認する		
	9		○		○	○		認可制御機能	役割に応じたアクセス権を付与するできるようにする	アカウントの役割に応じたアクセス権を付与する機能を持たせる	認可されていない情報や機能が利用できないことを確認する		
	10		○		○	○		サービス連携	ログイン情報が必要以上に他のサービスに渡らないようにする	サービス連携時に他のサービスに渡す情報をユーザに明示する機能を持たせる	他のサービスに渡した情報が何かを確認する		
II	1				○	○	ソフトウェア管理	Webアプリケーションファイアウォール	Webアプリケーションファイアウォールを利用できるようにする	Webアプリケーションファイアウォールを利用できるようにする	Webアプリケーションファイアウォールが利用できることを確認する		
	2		○		○	○		製品に含まれるファイアウォール機能	製品に含まれるファイアウォール機能を利用し、よりセキュアな状態にする	製品に含まれるファイアウォール機能を利用できるようにする	製品に含まれるファイアウォール機能が利用できることを確認する		
	3		○		○	○		ソフトウェアバージョン	脆弱性やバグ等に対応したバージョンのソフトウエアを利用し、セキュアな状態にしておく	ソフトウエアのアップデートを行う機能とバージョンを確認できる機能を持たせる	ソフトウエアのアップデートとバージョン確認ができることを確認する		
	4		○		○	○		ウィルス対策機能	製品に含まれるウイルス対策機能を利用し、よりセキュアな状態にする	製品に含まれるウイルス対策機能を利用できるようにする	製品に含まれるウイルス対策機能が利用できることを確認する		
	5		○		○	○		不正なデータ処理	システムが意図しない動作をしないようにする	受け付けるデータを制限する機能を持たせる	-		
	6		○	○	○	○		データ転送量	システムのデータ転送量はDDoS攻撃などを考慮した設計にする	受け付けるデータ転送量の制限を行うなどの機能を持たせる	受け付けるデータ転送量を確認する		
III	1		○		○	○	セキュリティ管理	ログ管理機能	インシデント発生時等に事態を把握するために、ログを保存する	システムに対して発生したイベントを記録するためにログ管理機能を持たせる	ログ情報が見られることを確認する		
	2		○		○	○		セッション管理 (Cookie設定)	システムでCookieを利用する場合、適切な属性を付与する	Cookieの適切な値にsecure属性、HttpOnly属性等を設定する	Cookieの適切な値にsecure属性、HttpOnly属性が設定されていることを確認する		
	3		○		○	○		セッション管理 (URLリライティング)	不必要なURLリライティングによってセッションIDが漏れないようにする	URLリライティングが必要かどうかを確認する	URLにセッションIDが埋め込まれていないか確認する		
	4		○		○	○		セッション管理 (ログイン時や重要な確定処理の時のセッションIDの払い出し)	セッション情報を取られることによる、機密情報の窃取のリスクを低減する	ログイン時や重要な確定処理の後に新しいセッションIDが発行され、古いセッションIDは破棄される実装にする	ログイン時や重要な確定処理の前後でセッションIDが変わっていることを確認する		
	5		○		○	○		クライアントデータの操作のセキュリティ対策	他のアカウントのデータを操作・閲覧できないようにする	アカウントごとのデータ管理機能を持たせる	他のアカウントのデータが操作・閲覧できないことを確認する		
	6		○		○	○		システムデータの操作のセキュリティ対策	システムデータは制限されたユーザのみが操作・閲覧できるようにする	特定のシステム管理者のみシステムデータの操作・閲覧ができる機能を持たせる	特定のシステム管理者以外でシステムデータが 操作・閲覧できないことを確認する		
	7				○	○		クラウドインタフェースやネットワークの脆弱性（APIインタフェースやクラウドベースのWebインタフェースなど）	クラウドインタフェースやネットワークの既知の脆弱性がシステムに存在しないかを確認する	公開情報を元に脆弱性情報を確認する、利用しているサービスをユーザに明示する機能を持たせる	公開情報を元に脆弱性情報を確認する		
	8				○	○		XSS、SQLi、およびCSRFの脆弱性	利用しているシステムにXSS、SQLi、CSRF等の既知の脆弱性が存在しないかを確認する	セキュアコーディングを意識し、XSS、SQLi、CSRF等の脆弱性を作りこまないようにする	公開情報を元に脆弱性情報を確認する		
	9				○	○		WebアプリケーションのSSL証明書	SSL実装の為の証明書を自身のシステムに適した形で実装する	システムに適した証明書を待たせる (例：EV認証証明書など)	利用している証明書を確認する		
IV	1		○		○	○	アクセス制御	管理されていない物理手段によるアクセス	管理されていない物理手段によるシステムへのアクセスを防ぐ	利用用途に応じてシステムにアクセスできる物理的手段(USB等)の制限を行える機能を持たせる	管理されていない物理的手段によるアクセスに対して制限ができているか確認する		
	2		○		○	○		リモートアクセス用ポートのデフォルトポート	リモートアクセス機能のデフォルトポートを狙った攻撃を防ぐ	デフォルトポートを変更するための機能を持たせる	デフォルトポートの変更を行えるか確認する		
	3		○	○	○	○		無線通信におけるセキュリティ(暗号化方式)	脆弱性を利用した通信内容の窃取を防ぐため、利用する暗号化方式はセキュアなものにする	セキュアな暗号化方式を利用できるようにする	接続時にセキュアな暗号化方式が選択されていることを確認する		
	4		○	○	○	○		無線通信におけるセキュリティ(WPS)	無線の設定ミスによるセキュリティの低下を防ぐ	WPS機能を持たせる場合はセキュリティを考慮する（例：MACアドレスフィルタリングなど）	WPSが動作するか確認する		
V	1		○		○	○	不正な接続	ネットワークポートの制限	利用用途を想定して、適切なポートのみを使えるようにする	利用用途に応じてポートの開閉を設定できる機能を持たせる	ポートの制御が設定したとおりに開閉されていることを確認する		
	2		○		○	○		UPnP	利用を想定しているデバイスに対して、UPnPが使えるようにする	UPnPの有効／無効を切り替える機能を持たせる	デバイスを接続したときに、設定した通りの挙動になっていることを確認する		
VI	1		○	○	○	○	暗号化	データの暗号化機能	データが平文で送られることにより、通信内容を読み取られることがないようにする	データを個別に暗号化する機能を持たせる	データを暗号化する機能があることを確認する		
	2		○	○	○	○		通信の暗号化機能	データが平文で送られることにより、通信内容を読み取られることがないようにする	システムを構成する機器間でSSL/TLSなどを利用した暗号化通信を行うための機能を持たせる	暗号化通信が利用できるようことを確認する		
	3		○	○	○	○		暗号化方式	利用する暗号化方式が確認できるようにする	利用する暗号化方式が確認できる機能を持たせる	利用している暗号化方式を確認する		
	4		○	○	○	○		証明書更新機能	証明書の期限が切れないようにする	証明書を更新するための機能を持たせる	証明書が有効であることを確認する		
VII	1	○					システム設定	センサーの動作状況確認機能	動作状況を確認できるようにする	現在のセンサーの動作状況を確認もしくは通知させる機能を持たせる	センサーの動作状況を確認する		
	2		○		○	○		ログのセキュリティ管理	第三者からログが閲覧されたり改ざんされたりすることを防止する	ログについて、閲覧可能なユーザの設定および内容の改ざんを防止する機能を持たせる	閲覧権限のないユーザでログが見れないことを確認する、閲覧可能なユーザでログが書き換えられないことを確認する		
VIII	1	○					通知	セキュリティイベントのアラートと通知機能 （状態異常等）	セキュリティイベント発生時にアラートを通知することにより、迅速に対応できるようにする	セキュリティイベント発生時のアラートを通知する機能を持たせる （例：状態異常等）	仕様通りに動作するか確認する		
	2		○		○	○		セキュリティイベントのアラートと通知機能 （認証失敗、証明書の期限切れ等）	セキュリティイベント発生時にアラートを通知することにより、迅速に対応できるようにする	セキュリティイベント発生時のアラートを通知する機能を持たせる （例：認証失敗、証明書の期限切れ等）	仕様通りに動作するか確認する		

Primitive凡例 S:Sensor、A:Aggregator、C:Communication Channel、E:e-Utility、D:Decision trigger

	No.	Primitive S	A	C	E	D	大項目	小項目	本項目の目的	開発する際に確認する項目	利用する際に確認する項目	回答欄	回答の補足
I	1	○	○		○	○	ユーザ管理	アカウントロックアウトメカニズム	第三者が端末を不正に操作できないようにする	連続した規定回数以上のログイン失敗や多重ログインなどの痕跡を確認したら、アカウントをロックし、ログインが不可になる機能を持たせる	アカウントロックに関する設定可能な内容を確認し、自身で設定したとおりにアカウントがロックされるか確認する		
	2		○		○	○		一定期間利用されていないアカウントの強制失効オプション	一定期間利用されていないアカウントからのログインをできないようにする	設定した有効期限を超過したアカウントをロックする機能を持たせる	有効期限後にアカウントが失効することを確認する		
	3		○		○	○		パスワード強度の担保機能	ブルートフォース、辞書攻撃などにより不正にログインされないようにする	複数の文字種の利用や一定文字数以上などの条件を満たしたパスワードのみ登録できる機能を持たせる	条件に満たないパスワードの登録ができないことを確認する		
	4		○		○	○		パスワードセキュリティオプション（二要素認証など）	第三者がシステムにログインすることを困難にする	パスワードセキュリティオプションを利用できるようにする （例：二要素認証など）	パスワードセキュリティオプションが利用できることを確認する		
	5		○		○	○		サービスやプロセスを起動するアカウントの権限管理	アカウント毎にサービスやプロセスを動かす権限を限定してインシデント発生時の影響範囲をサービスやプロセスの範囲内におさえる	サービスやプロセスの起動にスーパーユーザを求めない作りにする	サービスやプロセス専用のユーザで動作することを確認する		
	6		○		○	○		共有ユーザアカウント	用途に応じて適切な権限を付与できるようにする	アカウント管理機能を持たせる	共有するアカウントが適切な権限と共有範囲で利用されていることを確認する		
	7		○		○	○		管理ユーザへの適切な権限付与	管理ユーザが必要な権限を使えるようにする	権限を管理する機能を持たせる	管理者ユーザが適切な権限を付与されているか確認する		
	8		○		○	○		一般ユーザへの権限付与機能	ユーザに必要な権限のみを使えるようにする	ユーザに権限を付与できる機能を持たせる	ユーザに権限が付与でき、権限に応じた利用ができることを確認する		
	9		○		○	○		認可制御機能	役割に応じたアクセス権を付与するできるようにする	アカウントの役割に応じたアクセス権を付与する機能を持たせる	認可されていない情報や機能が利用できないことを確認する		
	10		○		○	○		サービス連携	ログイン情報が必要以上に他のサービスに渡らないようにする	サービス連携時に他のサービスに渡す情報をユーザに明示する機能を持たせる	他のサービスに渡した情報が何かを確認する		

Primitive凡例 S:Sensor、A:Aggregator、C:Communication Channel、E:e-Utility、D:Decision trigger

No.		Primitive					大項目	小項目	本項目の目的	開発する際に確認する項目	利用する際に確認する項目	回答欄	回答の補足
		S	A	C	E	D							
II	1				○	○	ソフトウェア管理	Webアプリケーションファイアウォール	Webアプリケーションファイアウォールを利用できるようにする	Webアプリケーションファイアウォールを利用できるようにする	Webアプリケーションファイアウォールが利用できることを確認する		
	2		○		○	○		製品に含まれるファイアウォール機能	製品に含まれるファイアウォール機能を利用し、よりセキュアな状態にする	製品に含まれるファイアウォール機能を利用できるようにする	製品に含まれるファイアウォール機能が利用できることを確認する		
	3		○		○	○		ソフトウェアバージョン	脆弱性やバグ等に対応したバージョンのソフトウエアを利用し、セキュアな状態にしておく	ソフトウエアのアップデートを行う機能とバージョンを確認できる機能を持たせる	ソフトウエアのアップデートとバージョン確認ができることを確認する		
	4		○		○	○		ウィルス対策機能	製品に含まれるウイルス対策機能を利用し、よりセキュアな状態にする	製品に含まれるウイルス対策機能を利用できるようにする	製品に含まれるウイルス対策機能が利用できることを確認する		
	5		○		○	○		不正なデータ処理	システムが意図しない動作をしないようにする	受け付けるデータを制限する機能を持たせる	-		
	6		○	○	○	○		データ転送量	システムのデータ転送量はDDoS攻撃などを考慮した設計にする	受け付けるデータ転送量の制限を行うなどの機能を持たせる	受け付けるデータ転送量を確認する		

Primitive凡例 S:Sensor、A:Aggregator、C:Communication Channel、E:e-Utility、D:Decision trigger

No.		Primitive					大項目	小項目	本項目の目的	開発する際に確認する項目	利用する際に確認する項目	回答欄	回答の補足
		S	A	C	E	D							
III	1		○		○	○	セキュリティ管理	ログ管理機能	インシデント発生時等に事態を把握するために、ログを保存する	システムに対して発生したイベントを記録するためにログ管理機能を持たせる	ログ情報が見られることを確認する		
	2		○		○	○		セッション管理 (Cookie設定)	システムでCookieを利用する場合、適切な属性を付与する	Cookieの適切な値にsecure属性、HttpOnly属性等を設定する	Cookieの適切な値にsecure属性、HttpOnly属性が設定されていることを確認する		
	3		○		○	○		セッション管理 (URLリライティング)	不必要なURLリライティングによってセッションIDが漏れないようにする	URLリライティングが必要かどうかを確認する	URLにセッションIDが埋め込まれていないか確認する		
	4		○		○	○		セッション管理 (ログイン時や重要な確定処理の時のセッションIDの払い出し)	セッション情報を取られることによる、機密情報の窃取のリスクを低減する	ログイン時や重要な確定処理の後に新しいセッションIDが発行され、古いセッションIDは破棄される実装にする	ログイン時や重要な確定処理の前後でセッションIDが変わっていることを確認する		
	5		○		○	○		クライアントデータの操作のセキュリティ対策	他のアカウントのデータを操作・閲覧できないようにする	アカウントごとのデータ管理機能を待たせる	他のアカウントのデータが操作・閲覧できないことを確認する		
	6		○		○	○		システムデータの操作のセキュリティ対策	システムデータは制限されたユーザのみが操作・閲覧できるようにする	特定のシステム管理者のみシステムデータの操作・閲覧ができる機能を持たせる	特定のシステム管理者以外でシステムデータが 操作・閲覧できないことを確認する		
	7				○	○		クラウドインタフェースやネットワークの脆弱性（APIインタフェースやクラウドベースのWebインタフェースなど）	クラウドインタフェースやネットワークの既知の脆弱性がシステムに存在しないかを確認する	公開情報を元に脆弱性情報を確認する、利用しているサービスをユーザに明示する機能を持たせる	公開情報を元に脆弱性情報を確認する		
	8				○	○		XSS、SQLi、およびCSRFの脆弱性	利用しているシステムにXSS、SQLi、CSRF等の既知の脆弱性が存在しないかを確認する	セキュアコーディングを意識し、XSS、SQLi、CSRF等の脆弱性を作りこまないようにする	公開情報を元に脆弱性情報を確認する		
	9				○	○		WebアプリケーションのSSL証明書	SSL実装の為の証明書を自身のシステムに適した形で実装する	システムに適した証明書を待たせる (例：EV認証証明書など)	利用している証明書を確認する		

Primitive凡例 S:Sensor、A:Aggregator、C:Communication Channel、E:e-Utility、D:Decision trigger

No.		Primitive					大項目	小項目	本項目の目的	開発する際に確認する項目	利用する際に確認する項目	回答欄	回答の補足
		S	A	C	E	D							
IV	1		○		○	○	アクセス制御	管理されていない物理手段によるアクセス	管理されていない物理手段によるシステムへのアクセスを防ぐ	利用用途に応じてシステムにアクセスできる物理的手段(USB等)の制限を行える機能を持たせる	管理されていない物理的手段によるアクセスに対して制限ができているか確認する		
	2		○		○	○		リモートアクセス用ポートのデフォルトポート	リモートアクセス機能のデフォルトポートを狙った攻撃を防ぐ	デフォルトポートを変更するための機能を持たせる	デフォルトポートの変更を行えるか確認する		
	3		○	○	○	○		無線通信におけるセキュリティ(暗号化方式)	脆弱性を利用した通信内容の窃取を防ぐため、利用する暗号化方式はセキュアなものにする	セキュアな暗号化方式を利用できるようにする	接続時にセキュアな暗号化方式が選択されていることを確認する		
	4		○	○	○	○		無線通信におけるセキュリティ(WPS)	無線の設定ミスによるセキュリティの低下を防ぐ	WPS機能を持たせる場合はセキュリティを考慮する（例：MACアドレスフィルタリングなど）	WPSが動作するか確認する		

Primitive凡例 S:Sensor、A:Aggregator、C:Communication Channel、E:e-Utility、D:Decision trigger

No.		Primitive					大項目	小項目	本項目の目的	開発する際に確認する項目	利用する際に確認する項目	回答欄	回答の補足
		S	A	C	E	D							
V	1		○		○	○	不正な接続	ネットワークポートの制限	利用用途を想定して、適切なポートのみを使えるようにする	利用用途に応じてポートの開閉を設定できる機能を持たせる	ポートの制御が設定したとおりに開閉されていることを確認する		
	2		○		○	○		UPnP	利用を想定しているデバイスに対して、UPnPが使えるようにする	UPnPの有効／無効を切り替える機能を持たせる	デバイスを接続したときに、設定した通りの挙動になっていることを確認する		

Primitive凡例 S:Sensor、A:Aggregator、C:Communication Channel、E:e-Utility、D:Decision trigger

No.		Primitive					大項目	小項目	本項目の目的	開発する際に確認する項目	利用する際に確認する項目	回答欄	回答の補足
		S	A	C	E	D							
VI	1		○	○	○	○	暗号化	データの暗号化機能	データが平文で送られることにより、通信内容を読み取られることがないようにする	データを個別に暗号化する機能を持たせる	データを暗号化する機能があることを確認する		
	2		○	○	○	○		通信の暗号化機能	データが平文で送られることにより、通信内容を読み取られることがないようにする	システムを構成する機器間でSSL/TLSなどを利用した暗号化通信を行うための機能を持たせる	暗号化通信が利用できるようことを確認する		
	3		○	○	○	○		暗号化方式	利用する暗号化方式が確認できるようにする	利用する暗号化方式が確認できる機能を持たせる	利用している暗号化方式を確認する		
	4		○	○	○	○		証明書更新機能	証明書の期限が切れないようにする	証明書を更新するための機能を持たせる	証明書が有効であることを確認する		

Primitive凡例 S:Sensor、A:Aggregator、C:Communication Channel、E:e-Utility、D:Decision trigger

No.		Primitive S	A	C	E	D	大項目	小項目	本項目の目的	開発する際に確認する項目	利用する際に確認する項目	回答欄	回答の補足
VII	1	○					システム設定	センサーの動作状況確認機能	動作状況を確認できるようにする	現在のセンサーの動作状況を確認もしくは通知させる機能を持たせる	センサーの動作状況を確認する		
	2		○		○	○		ログのセキュリティ管理	第三者からログが閲覧されたり改ざんされたりすることを防止する	ログについて、閲覧可能なユーザの設定および内容の改ざんを防止する機能を持たせる	閲覧権限のないユーザでログが見れないことを確認する、閲覧可能なユーザでログが書き換えられないことを確認する		

Primitive凡例 S:Sensor、A:Aggregator、C:Communication Channel、E:e-Utility、D:Decision trigger

No.		Primitive					大項目	小項目	本項目の目的	開発する際に確認する項目	利用する際に確認する項目	回答欄	回答の補足
		S	A	C	E	D							
VIII	1	○					通知	セキュリティイベントのアラートと通知機能 （状態異常等）	セキュリティイベント発生時にアラートを通知することにより、迅速に対応できるようにする	セキュリティイベント発生時のアラートを通知する機能を持たせる （例：状態異常等）	仕様通りに動作するか確認する		
	2		○		○	○		セキュリティイベントのアラートと通知機能 （認証失敗、証明書の期限切れ等）	セキュリティイベント発生時にアラートを通知することにより、迅速に対応できるようにする	セキュリティイベント発生時のアラートを通知する機能を持たせる （例：認証失敗、証明書の期限切れ等）	仕様通りに動作するか確認する		

Primitive（IoTシステムを構成する基本単位）構成要素

Sensor	温度、加速度、重量、音、位置などを測定する機能・機器
Aggregator	センサからのデータを集約する機能・機器
Communication Channel	データの送受信を行うための通信路・ネットワーク
e-Utility	データを閲覧したり設定したりするインタフェース
Decision Trigger	データを計算し、その結果に基づいてアクションさせるための機能

JPCERT Coordination Center