USecurD

# API der USecureD-Tools

## Deliverable E3

| | |
|---|---|
| Projekt | USecureD – Usable Security by Design |
| Förderinitiative | Einfach intuitiv – Usability für den Mittelstand |
| Förderkennzeichen | 01MU14002 |
| Arbeitspaket | AP 3 |
| Fälligkeit | 31.10.2016 |
| Autoren | Luigi Lo Iacono (TH Köln) |
| | Peter Nehren (TH Köln) |
| Status | Entwurf |
| Klassifikation | Intern |


HKBS

**HK Business Solutions GmbH**
Hartmut Schmitt
Mellinweg 20
66280 Sulzbach
schmitt@hk-bs.de

KMU
(Konsortialführer)

Technology
Arts Sciences
TH Köln

**Technische Hochschule Köln**
Prof. Dr.-Ing. Luigi Lo Iacono
Betzdorfer Straße 2
50679 Köln
luigi.lo_iacono@th-koeln.de

Hochschule
(Konsortialpartner)



Mittelstand-Digital

Gefördert durch:
Bundesministerium für Wirtschaft und Energie
aufgrund eines Beschlusses des Deutschen Bundestages

## Abstract

Die in Arbeitspaket 2.1 vorgestellten USecureD-Werkzeuge sind zunächst nur über die entwickelte webbasierte Plattform aufrufbar gewesen (https://das.th-koeln.de/usecured). Durch Anregungen externer Anwender, die die USecureD-Tools bereits kurz nach der Veröffentlichung verwendet haben, wurde ein Bedarf für eine Programmierschnittstelle zum automatisierbaren maschinellen Abruf der Werkzeuge festgestellt. Aus diesem Grund ist eine REST-basierende API entwickelt worden, die die verschiedenen Artefakte in einem spezifizierten Datenformat basierend auf der Beschreibungssprache JSON bereitstellt. Eine ausführliche Beschreibung der Programmierschnittstelle (engl. Application Programming Interface, API) ist in einer technischen Spezifikation in der OpenAPI Beschreibungssprache (https://www.openapis.org/) auf der Plattform veröffentlicht worden. Diese Schnittstelle ermöglicht es Unternehmen, die USecureD-Werkzeuge automatisiert in eigene Prozesse und Produkte bzw. Systeme zu integrieren und z.B. für Qualitätserhebungen oder Assessments einzusetzen. Im Folgenden werden die entwickelte Datenformate und die API-Spezifikation exemplarisch dargelegt und mit den jeweils vollständigen technischen Spezifikationen verlinkt.



## Schlagworte

Werkzeuge, Programmierschnittstelle, REST, REST-API, OpenAPI, maschineller Abruf, JSON, YAML, Swagger

# Inhaltsverzeichnis

# 1 Vorgehensweise

Die in AP 2.1 entwickelten Werkzeuge werden sowohl intern, als auch extern evaluiert, um die Funktionalität und Gebrauchstauglichkeit dieser zu überprüfen. Dies ermöglicht es Verbesserungen aufgrund von angebrachtem Feedback umzusetzen und so die Funktionalität und Gebrauchstauglichkeit der Plattform bzw. Werkzeuge noch während der Projektlaufzeit zu verbessern. Beispielsweise hat sich nach der Veröffentlichung der Werkzeuge auf der USecureD-Plattform herausgestellt, dass ein programmatischer Zugriff auf die Werkzeuge über ein REST-basierenden API für einige Anwender von großem Nutzen ist. Dies war der Auslöser um zu analysieren, wie eine solche API in die aktuelle Version der webbasierten USecureD-Tools integriert werden und welche Schritte für die Umsetzung notwendig sind.

Als erster Schritt ist das Datenbeschreibungsformat JSON[1] (JavaScript Object Notation) als zusätzliche Repräsentation der Werkzeuge – neben HTML – festgelegt worden. Mit JSON können Daten leichtgewichtig beschrieben werden, wodurch das textbasierte Datenformat insgesamt eine weite Verbreitung gefunden hat. Ein Teil der Software der USecureD-Plattform ist anschließend umstrukturiert und erweitert worden, sodass bei der Generierung der Werkzeuge neben der HTML-Repräsentation zusätzlich eine dazugehörige JSON-Repräsentation erstellt wird. Diese enthält alle Informationen und Verknüpfungen, welche auch in den Beschreibungsvorlagen der Werkzeuge und auf der Plattform in HTML zu sehen sind (vom Browser gerendert). Darüber hinaus sind JSON-Repräsentationen für eine Liste aller Prinzipien, Richtlinen und Patterns erstellt worden, in welcher die Namen und die Verlinkungen zu den einzelnen Werkzeugen enthalten sind. Wie für REST-basierende Web APIs üblich wird über den *Accept*-Header[2] bestimmt, welche Repräsentation zurückgeliefert wird (z.B. Accept: application/json für JSON). Über den *Accept-Language*-Header kann zudem die Sprache des angeforderten Werkzeugs eingestellt werden (aktuell Deutsch und Englisch).

Eine Service-Beschreibung[3] der gesamten REST-ful Web API kann auf der Plattform eingesehen werden. Diese wurde mit Hilfe des API-Beschreibungs-Frameworks Swagger[4] erstellt, wodurch Entwickler einen schnellen Überblick über die Funktionalität erlangen und auch Programmteile zum Abruf der Werkzeuge in verschiedenen Sprachen automatisiert erstellt werden können. Diese Service-Beschreibung mit Swagger macht es Unternehmen möglich, die im USecureD-Projekt erarbeiteten Werkzeuge in eigene Entwicklungs- oder Evaluierungsprozesse zu integrieren und z.B. für Qualitätserhebungen oder Assessments zu verwenden.

[1] http://www.json.org/json-de.html
[2] https://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html
[3] API Beschreibung erreichbar unter: https://das.th-koeln.de/usecured/assets/api/swagger.yaml (oder .json)
[4] http://swagger.io/

# 2 Exemplarisches JSON-Dokument eines Usable Security Patterns

Das nachfolgende JSON-Dokument zeigt exemplarisch die entwickelte Datenstruktur, welche ein Usable Security Pattern in Deutsch dokumentiert. An dem Beispiel kann die generelle Struktur nachvollzogen und der Bezug zum Patterntemplate (siehe Deliverable E2.1) hergestellt werden.


```
{
  "metadata": {
    "log": "[03/01/2016]: Translated to German",
    "translations": [
      "en"
    ],
    "lang": "de",
    "type": "usable security pattern"
  },
  "references": [
    {
      "Egelman11": {
        "description": "Egelman, S., 2009. Trust Me: Design Patterns for Constructing Trustworthy Trust Indicators. ProQuest.",
        "url": null
      }
    }
  ],
  "synonyms": null,
  "implementation": "Aktive Warnungen müssen den Nutzer aus seiner Primäraufgabe herausreißen und dafür entweder den Inhalt, den der Nutzer erwartet, mit einer Warnung ersetzen oder sie müssen die Aufmerksamkeit weg von dem erwarteten Inhalt lenken.",
  "check lists": null,
  "dependencies": null,
  "tags": "Active Warnings, Immediate Notifications, Warnings",
  "relationships": [
    {
      "id": "attractive-options",
      "url": "https://das.th-koeln.de/usecured/patterns/attractive-options"
    },
    {
      "id": "immediate-notifications",
      "url": "https://das.th-koeln.de/usecured/patterns/immediate-notifications"
    },
    {
      "id": "conveying-threats-and-consequences",
      "url": "https://das.th-koeln.de/usecured/patterns/conveying-threats-and-consequences"
    },
    {
      "id": "general-notifications-about-security",
      "url": "https://das.th-koeln.de/usecured/patterns/general-notifications-about-security"
    },
    {
      "id": "immediate-options",
      "url": "https://das.th-koeln.de/usecured/patterns/immediate-options"
    },
    {
      "id": "separating-content",
      "url": "https://das.th-koeln.de/usecured/patterns/separating-content"
```

```
    }
  ],
  "problem": "Manche Warnungen schlagen in kritischen Situationen fehl, weil diese nicht aufdringlich genug sind und der Nutzer sie nicht bemerkt.",
  "solution": "Aktive Warnungen sollen die Aufmerksamkeit des Nutzers auf sich lenken, in dem sie ihn aus seiner Primäraufgabe herausreißen. Somit wird der Nutzer zum Anerkennen der Warnung gezwungen und kann darauf reagieren.",
  "examples": {
    "image": "https://das.th-koeln.de/usecured/patterns/images/active-warnings.png",
    "description": "Quelle: (Egelman, 2009)"
  },
  "use cases": null,
  "name": "Aktive Warnungen",
  "principles": [
    {
      "id": "convenience",
      "url": "https://das.th-koeln.de/usecured/principles/convenience"
    },
    {
      "id": "clarity",
      "url": "https://das.th-koeln.de/usecured/principles/clarity"
    }
  ],
  "consequences": "Der Nutzer wird in seiner Primäraufgabe unterbrochen und muss eine Entscheidung treffen. Dadurch werden wesentlich mehr Warnungen beachtet und der Nutzer im Endeffekt besser vor Phishing-Attacken geschützt.",
  "guidelines": null,
  "id": "active-warnings"
}
```

# 3 Vollständige Spezifikation der API

Die programmatische Schnittstelle wurde als REST API entwickelt. Die Spezifikation der REST API ist gemäß der OpenAPI Spezifikation (https://openapis.org/) erfolgt. Die vollständige Spezifikation ist im Folgenden angegeben. Die Beschreibung enthält alle Endpunkte und dokumentiert wie der Aufruf dieser erfolgen muss einschließlich der benötigten Eingabeparameter und der zurückgelieferten Rückgabeparameter. Entwickler können diese unter der Adresse https://das.th-koeln.de/usecured/assets/api/swagger.yaml abrufen.

```
swagger: '2.0'

info:
  version: "1.0.0"
  title: USecureD API
  description: RESTful HTTP API providing usable security principles, guidelines and patterns
  contact:
    name: USecureD
    url: https://www.usecured.de
    email: peter.nehren@th-koeln.de
host: das.th-koeln.de
basePath: /usecured
# USecureD API paths
paths:
  /patterns:
    get:
      description: |
        Get list of all patterns
      parameters:
        - in: header
          name: accept
          description: HTML (text/html) and JSON (application/json) are supported so far
          required: false
          type: string
        - in: header
          name: accept-language
          description: Englisch (en) and German (de) are supported so far
          required: false
          type: string
      # Expected responses for this operation:
      produces:
              - text/html (default)
              - application/json
      responses:
        # Response code
        200:
          description: Successful response
          schema:
            title: ArrayOfPatterns
            type: array
            items:
              title: Pattern
              type: object
              properties:
                name:
                  type: string
                id:
                  type: string
                url:
                  type: string

  /patterns/{id}:
    get:
      description: Retrieve a single pattern by its ID (IDs can be obtained from list of all
patterns)
      parameters:
              - in: header
                name: accept
                description: HTML (text/html) and JSON (application/json) are supported so far
                required: false
                type: string
              - in: header
                name: accept-language
                description: Englisch (en) and German (de) are supported so far
                required: false
                type: string
              - name: id
```

```
          in: path
          description: ID pattern
          required: true
          type: string
produces:
  - text/html (default)
  - application/json
responses:
  200:
    description: pattern response
    schema:
      title: Pattern
      type: object
      properties:
        name:
          type: string
        id:
          type: string
        problem:
          type: string
        context:
          type: string
        consequences:
          type: string
        solution:
          type: string
        implementation:
          type: string
        synonyms:
          type: string
        references:
          type: array
          items:
              title: References
              type: object
              properties:
                additionalProperties:
                  properties:
                    description:
                      type: string
                    url:
                      type: string
        relationships:
          type: array
          items:
              title: Relationships
              type: object
              properties:
                id:
                  type: string
                url:
                  type: string
        examples:
          type: object
          properties:
            image:
              type: string
            description:
              type: string
        tags:
          type: string
        use cases:
          type: string
        check lists:
          type: string
        metadata:
         type: object
         properties:
            log:
              type: string
            lang:
              type: string
            type:
              type: string
            translations:
              type: array
              items:
                type: string
```

```
  /principles:
    get:
      description: |
        Get list of all principles
      parameters:
              - in: header
                name: accept
                description: HTML (text/html) and JSON (application/json) are supported so far
                required: false
                type: string
              - in: header
                name: accept-language
                description: Englisch (en) and German (de) are supported so far
                required: false
                type: string
      # Expected responses for this operation:
      produces:
              - text/html (default)
              - application/json
      responses:
        # Response code
        200:
          description: Successful response
          schema:
            title: ArrayOfPrinciples
            type: array
            items:
              title: Principle
              type: object
              properties:
                name:
                  type: string
                id:
                  type: string
                url:
                  type: string

  /principles/{id}:
    get:
      description: Retrieve a single principle by its ID (IDs can be obtained from list of all
principles)
      parameters:
              - in: header
                name: accept
                description: HTML (text/html) and JSON (application/json) are supported so far
                required: false
                type: string
              - in: header
                name: accept-language
                description: Englisch (en) and German (de) are supported so far
                required: false
                type: string
              - name: id
                in: path
                description: ID principle
                required: true
                type: string
      produces:
        - text/html (default)
        - application/json
      responses:
        200:
          description: principle response
          schema:
            title: Principle
            type: object
            properties:
              name:
                type: string
              id:
                type: string
              motivation:
                type: string
              intention:
                type: string
              synonyms:
```

```
              type: string
            references:
              type: array
              items:
                  title: References
                  type: object
                  properties:
                    additionalProperties:
                      properties:
                        description:
                          type: string
                        url:
                          type: string
            guidelines:
              type: array
              items:
                  title: Guidelines
                  type: object
                  properties:
                    id:
                      type: string
                    url:
                      type: string
            examples:
              type: object
              properties:
                image:
                  type: string
                description:
                  type: string
            tags:
              type: string
            metadata:
             type: object
             properties:
                log:
                  type: string
                lang:
                  type: string
                type:
                  type: string
                translations:
                  type: array
                  items:
                    type: string

/guidelines:
  get:
    description: |
      Get list of all guidelines
    parameters:
            - in: header
              name: accept
              description: HTML (text/html) and JSON (application/json) are supported so far
              required: false
              type: string
            - in: header
              name: accept-language
              description: German (de) is supported so far
              required: false
              type: string
    produces:
            - text/html (default)
            - application/json
    responses:
      # Response code
      200:
        description: Successful response
        schema:
          title: ArrayOfGuidelines
          type: array
          items:
            title: Guideline
            type: object
            properties:
              name:
                type: string
              id:
```

```
                type: string
              url:
                type: string

  /guidelines/{id}:
    get:
      description: Retrieve a single guideline by its ID (IDs can be obtained from list of all
guidelines)
      parameters:
              - in: header
                name: accept
                description: HTML (text/html) and JSON (application/json) are supported so far
                required: false
                type: string
              - in: header
                name: accept-language
                description: German (de) is supported so far
                required: false
                type: string
              - name: id
                in: path
                description: ID guideline
                required: true
                type: string
      produces:
        - text/html (default)
        - application/json
      responses:
        200:
          description: guideline response
          schema:
            title: Guideline
            type: object
            properties:
              name:
                type: string
              id:
                type: string
              solution:
                type: string
              guideline:
                type: string
              synonyms:
                type: string
              category:
                type: string
              references:
                type: array
                items:
                    title: References
                    type: object
                    properties:
                      additionalProperties:
                        properties:
                          description:
                            type: string
                          url:
                            type: string
              related guidelines:
                type: array
                items:
                    title: Related guidelines
                    type: object
                    properties:
                      id:
                        type: string
                      url:
                        type: string
              examples:
                type: object
                properties:
                  image:
                    type: string
                  description:
                    type: string
              tags:
                type: string
              metadata:
```

```
type: object
properties:
   log:
     type: string
   lang:
     type: string
   type:
     type: string
   translations:
     type: array
     items:
       type: string
```