



31-03-2015

Open Call Deliverable OCT-DS1.1

Final Report Secure Enterprise Networks Simple & Easy (SENSE)

Open Call Deliverable OCT-DS1.1

Grant Agreement No.: 605243

Activity: NA1

Task Item: 10

Nature of Deliverable: R (Report)

Dissemination Level: PU (Public)

Lead Partner: PSNC

Document Code: GN3PLUS14-1301-49

Authors: Stefan Winter, RESTENA (editor); Tomasz Wolniewicz, PSNC/NCU; Gareth Ayres, Swansea University; Michał Gasewicz, PSNC/NCU; Maja Górecka--Wolniewicz, PSNC/NCU; Jędrzej Jajor, PSNC; Tomasz Krakowski, PSNC/NCU; Zbigniew Ołtuszyk, PSNC; Łukasz Zygmąński, PSNC

© GEANT Limited on behalf of the GN3plus project.

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7 2007–2013) under Grant Agreement No. 605243 (GN3plus).

Abstract

This is the final deliverable of the SENSE Open Call project, which worked on the challenge of improving the imperfect EAP supplicant software landscape. It describes the project's outcomes (including, but not limited to: an Android application for proper enterprise Wi-Fi setup; a "roaming consortium sandbox" – the EAPlab; and a proposed new standard way for cross-platform configuration of enterprise networking), its impact in the enterprise network authentication industry niche, and possible future work in the problem space.

Table of Contents

Executive Summary	1
1 Introduction	2
2 Project Results	4
2.1 R&D Work (Work Package 2)	4
2.1.1 EAP Metadata Standardisation (Task 2.1)	4
2.1.2 Quality Evaluation and Improvement of Supplicants (Task 2.2)	6
2.1.3 EAP Lab (Task 2.3)	12
2.1.4 Generic Configurator Development (Task 2.4)	27
2.2 Management and Dissemination (Work Package 1)	32
2.2.1 Project Management (Task 1.1)	32
2.2.2 Dissemination of Project Results (Task 1.2)	33
3 Project Impact	37
4 Challenges	38
4.1 KDE User Interface Improvement	38
4.2 Network configuration beyond EAP	39
5 Mapping of results to the requirements of the call text and to the Open Call scope	41
5.1 Overall Open Call Scope	41
5.2 Specific Call Text of Open Call #16	42
5.3 Outlook	43
6 Conclusions	45
Appendix A Android eduroam CAT: IdP Admin Guide	47
A.1 Introduction	47
A.2 The eduroam CAT App	47
A.3 The eap-config File	52
A.4 Play Store and Captive Portals	52
A.5 Example: Swansea University	52

Table of Figures

Figure 1 – Install Section	48
Figure 2 – Profiles Section	49
Figure 3 – Status section	50
Figure 4 – Install Profile Section	51

Executive Summary

This document is the final deliverable of the SENSE project (“Secure Enterprise Networks Simple & Easy”), one of the 21 Open Call projects inside GN3plus. The deliverable describes the results of SENSE, all of which contribute to delivering the overall objective of the project: to improve enterprise Wi-Fi authentication – make the needed authentication software (“EAP supplicants”) on many platforms more secure, more user-friendly and more feature-rich.

The project delivered significant results and had an impact in all its four main work areas:

- (1) SENSE has developed a multi-platform configuration file format which is able to describe all authentication parameters for the EAP protocol and has pushed this file format to a standards-defining organisation (IETF)
- (2) SENSE has defined a metrics which allows the assessment of EAP supplicants in the three categories: security, user interface, feature set and has evaluated seven existing EAP supplicants against this metrics. Among the major findings of this task is the fact that very few supplicants excel, or even just do an acceptable job. Much more work in this space is needed in the future.
- (3) SENSE has created a toolkit for evaluation of many aspects of an enterprise Wi-Fi deployment: a sandbox roaming infrastructure, several RADIUS/EAP servers to authenticate against, and a web interface to provoke erroneous behaviour and to test corner cases. The toolkit, called “EAPlab”, is already now used beyond project boundaries by the industry at large.
- (4) SENSE has written and published two EAP supplicants to ameliorate the enterprise Wi-Fi authentication on platforms with a dire need: the Android operating system (in form of an “app” for the Google Play Store) and for the Linux operating system (where user-interactive configuration is possible with a variety of tools, but no automatic deployment without user intervention)

The further content of the deliverable summarises the project management (no issues), the dissemination activities which were performed (twice as many as planned for in the original proposal), enumerates the impacts of the project beyond the GÉANT space and makes a case for a future continuation of work in the same problem space.

SENSE has delivered significant value for money in its 18 months runtime. A continuation in a future round of Open Calls or as part of the regular GÉANT Innovation Programme is recommended.

1 Introduction

Part of the GN3plus project was a round of Open Calls. Proposals for a number of subjects were sought, and a total of 21 projects were selected for funding in the Open Calls scheme. SENSE (“Secure Enterprise Networks Simple & Easy”) is one of those projects. The proposal targeted the Open Call #16 “IEEE 802.1X and EAP – Improving Implementation Completeness and User-Friendliness” and had a total runtime of 18 months.

Its objective is to ensure that all important EAP methods are properly supported by the commonly used or popular platforms.

This is unfortunately not a given in today’s enterprise wireless authentication software on many platforms (commonly called the “EAP Supplicant”). Supplicants regularly miss out on essential security features, do not provide users with all options or required configuration settings needed for proper operation, and do not always provide helpful error messages when things go wrong.

SENSE has transformed this overarching objective of improving the EAP supplicant landscape into four specific work areas:

- EAP Metadata Standardisation (Task 2.1)
- Quality Evaluation and Improvement of Supplicants (Task 2.2)
- EAP Lab (Task 2.3)
- Generic Configurator Development (Task 2.4)

This document is the final deliverable of the project. It summarises all the development and dissemination activities undertaken during the project runtime. The goal of this document is to make readers aware of what was delivered in the project to enable a final assessment of the actual project outcomes versus the stated goals. The document assumes a certain familiarity of the reader with the problem space at hand.

The document contains the following chapters: chapter 2 “Project Results” describes the work undertaken in the project. It is followed by chapter 3 “Impact”, where the authors show how the project generated actual impact beyond the GÉANT environment. Naturally, a project of this size encounters unforeseen challenges that deserve attention and elaboration. Those challenges are described in chapter 4 “Challenges”. Chapter 5 “Mapping of results” then shows how the results of

this project fit into the wider environment of the Open Calls. With the project coming to an end, it also is important to look into the future – where should the work be continued, where is new work on the horizon which deserves being tackled in future projects? Chapter 6 “Outlook” provides this information. The document ends with “Conclusions” in chapter 7.

2 Project Results

The project's main work is organised within work package 2, with the exception of management and the dissemination of project results, which form work package 1. For logical coherence reasons, this deliverable first presents the results of the actually executed core R&D work in work package 2, and then describes work package 1's dissemination efforts as those naturally depend on said R&D work.

2.1 R&D Work (Work Package 2)

The R&D work in the project concentrated on four work areas, each organised in its own task:

- EAP Metadata Standardisation (Task 2.1)
- Quality Evaluation and Improvement of Supplicants (Task 2.2)
- EAP Lab (Task 2.3)
- Generic Configurator Development (Task 2.4)

The following four sections describe each of those tasks in detail.

2.1.1 EAP Metadata Standardisation (Task 2.1)

This section describes the task related to standardisation challenges. It is organised in three sub-sections: first the expected results as defined in the project plan, followed by a description of the work as it was actually carried out (explaining minor derivations where they became necessary), and finally a presentation of the state of work as it was at the end of the project.

2.1.1.1 *Expected Results per Project Plan*

The objective of this work package is described in the project plan as:

"To provide an industry standard way of expressing EAP configuration details, for consumption by EAP-capable devices."

The following milestones and KPIs relate to the task:

M2.1 File Format Stabilised - XML definition in a stable format – ready for other tasks to depend on it

KPI-3 Submit draft specification of the EAP metadata format to a standardization organisation

2.1.1.2 Approach and realisation of results during project runtime

The work plan of task 2.1 in the original project description was followed; the plan is not repeated here for brevity.

The work plan foresaw gathering feedback from the IETF. There was indeed substantial feedback: IETF experts criticised SENSE's original choice of using XML to describe EAP configuration details. The IETF has its own language to express network configuration, YANG; and the experts suggested to define EAP properties in YANG.

YANG is not typically consumed directly by end devices, but any configuration information specified in YANG can be transcribed into either XML or JSON without information loss; automatic conversion tools exist. Since client devices may prefer parsing and usage of one of those two formats, it becomes possible to satisfy these platform-specific requirements in an automated conversion.

This is indeed a significant positive argument towards using YANG, and consequently the second iteration of the draft specification is based on YANG. The only downside to this decision is that the derived XML Schema has significant differences to the native XML Schema that was developed for the initial draft specification.

Despite giving in to the arguments of the IETF, the overall interest in the specification has remained on a modest level. Several individuals during IETF meetings indicated their interest in this specification, but the inertia was not high enough to lead to official adoption in the pertinent working group yet.

At the same time, the significant XML Schema changes for the SENSE implementation work meant a good amount of work to do – which would potentially be wasted if the IETF decides not to standardise this specification.

So, it was decided to further pursue standardisation in the YANG-based specification, but to keep using the initial -00 version of the document for the implementations created inside SENSE. At a later point, when the specification becomes adopted by an official IETF working group or accepted for the so-called Independent Stream (official publication, without backing of a working group), updates to the implementations can be done; if the IETF venue turns out to be a dead end, no effort was wasted on the pursuit of this route.

2.1.1.3 Description of final result

There were altogether two iterations of the draft specification. The first version was submitted in time for the 89th IETF Meeting, which completes KPI-3, and is used as the stable version for the implementation work in task 2.4, thereby completing M2.1.

The specification is available at <http://tools.ietf.org/html/draft-winter-opsawg-eap-metadata-00> ; the XML Schema itself can be downloaded from <http://ticker.eduroam.lu/cat/EAP-metadata/eap-metadata.xsd> . The sources for the specification and the Schema are also available in the GÉANT SVN repository (folder GEANT -> sense -> EAP-metadata). The specification was presented at the 89th IETF meeting; see the Dissemination Results section for more details.

A second version was submitted for the 91st IETF meeting with the substantial differences to the first version as detailed in the previous section. The implementations that were written inside SENSE do not yet use the derived XML of this version. The specification can be found at <http://tools.ietf.org/html/draft-winter-opsawg-eap-metadata-01> , the YANG model description is at <https://www.suplicants.net/site/standardisation/eap-metadata-01.yang> and the automatically derived XML Schema for this YANG model can be found at <https://www.suplicants.net/site/standardisation/eap-metadata-01.xml> . This updated specification was presented at the 91st IETF meeting; see the Dissemination Results section for more details.

2.1.2 Quality Evaluation and Improvement of Supplicants (Task 2.2)

In this section, the work relating to supplicants is described: the definition of quality criteria and the subsequent assessment of devices against those criteria. It is organised in three sub-sections: first the expected results as defined in the project plan, followed by a description of the work as it was actually carried out (explaining minor derivations where they became necessary), and finally a presentation of the state of work as it was at the end of the project.

2.1.2.1 Expected Results per Project Plan

This task is expected to deliver the following objectives:

1. *"To define metrics which allow to assess an EAP supplicant regarding its user-interface design, its technical completeness, and its automated configurability."*
2. *"To evaluate a wide set of supplicants which are available on the market against these criteria".*

It is measured against the following KPIs:

- KPI-1 The project has defined at least 10 distinct criteria under which to assess the quality of supplicants
- KPI-2 The project has assessed at least 7 different supplicant implementations with the criteria above

2.1.2.2 *Approach and realisation of results during project runtime*

The approach taken for this task largely followed the initial sketch in the work plan description in the original project proposal, and is not repeated here. As the definition of criteria went along, it became apparent that minor adjustments to the plan would lead to a better result:

- Initially, the range of scores was envisaged to be positive or negative. The project determined that almost every criterion can be formulated positively or negatively (thereby inverting the “direction” of the scoring), so it is sufficient to define scores from 0 to a maximum score, where 0 reflects the bad outcome of the assessment, and a positive score reflects the good outcome of an assessment.
- The security parameters were planned to always be hard failures. In the course of the supplicant examination process, it became clear that while there are some knock-out criteria, there are also some desirable security properties which can be worked around if not present in the device. The security criteria were thus split into CRITICAL criteria, leading to hard failure (a “red card”); MAJOR criteria, leading to significant penalty (a “yellow card”); and MINOR criteria, which receive a score like all other non-security criteria.
- To enhance community engagement (among other reasons, as a reaction to the Evaluation Summary Report remark regarding lack of collaboration with the eduroam community world-wide), the weighting of the criteria was not done internally in the project, but sent to the pertinent eduroam community mailing list as a survey. Respondents to the survey could indicate how important they feel a certain criterion is, and whether criteria are superfluous or missing in the initial set. The result of the survey is that one criterion was promoted from “feature” to “major security” (see criterion 9 in the list below) and that all integer criteria were put into six classes of importance (mapping to the integer scores 1 to 6).
- The work plan foresaw that one specific supplicant user interface could optionally be improved by subcontracting its developer. This was attempted, but did not lead to significant improvements and no payments were made. For a detailed explanation, see the chapter “Challenges” below.

The work plan was executed with these modifications, and resulted in the following final project output:

2.1.2.3 *Description of final result*

Altogether, 32 distinct criteria for the security, usability and feature-completeness of EAP supplicants were defined. This surpasses the requirements as set forth in KPI-1. The following tables list:

- the criteria and the expected outcome, along with the category they belong in and their weighting
- the assessed devices, their performance in all the criteria, and the resulting overall score

NR	CATEGORY	DESCRIPTION	EXPECTED	CONSEQUENCE
1	SEC-CRITICAL	Can custom CAs be installed?	Yes	RED
2	SEC-CRITICAL	Can the trusted CA(s) be specified?	Yes	RED
3	SEC-CRITICAL	Does the device connect to the genuine network?	Yes	RED
4	SEC-CRITICAL	Are expired server certificates rejected?	Yes	RED
5	SEC-CRITICAL	Are server certificates from an untrusted CA rejected?	Yes	RED
6	SEC-CRITICAL	Are there severe device-specific security bugs?	No	RED
7	SEC-MAJOR	Is there a dedicated Wi-Fi trust store for CAs?	Yes	YELLOW
8	SEC-MAJOR	Can the trusted server name be specified?	Yes	YELLOW
9	SEC-MAJOR	Is a fallback to cert fingerprint server authentication available?	Yes	YELLOW
10	SEC-MINOR	Does the device check the CRL status of the server certificate?	Yes	5
11	SEC-MINOR	Can the user choose between encrypted vs. unprotected user certificate private key (beyond overall cert store locks, if any)?	Yes	3
12	SEC-MINOR	Is it possible to setup Enterprise Wi-Fi in a secure way on first use?	Yes	2
13	USABILITY	Does the setup of Enterprise Wi-Fi potentially introduce significant changes in the work flow of device usage?	No	5
14	USABILITY	Is the supplicant user interface overall comprehensible?	Yes	5
15	USABILITY	Does the supplicant erroneously make the system CA store the default choice for CAs?	No	6
16	USABILITY	Can the network configuration be altered after the initial setup?	Yes	5
17	USABILITY	Do non-interactive connection attempts fail non-interactively; and do interactive connection attempts fail with error information for the user?	Yes	4
18	USABILITY	Do occasional failed authentications lead to deletion of the configuration?	No	4
19	USABILITY	Can more than one CA be marked as trusted for an Enterprise network?	Yes	3

NR	CATEGORY	DESCRIPTION	EXPECTED	CONSEQUENCE
20	USABILITY	Does the correct configuration of Enterprise Wi-Fi trigger security warnings?	No	6
21	USABILITY	When failing to authenticate against the genuine authentication server, is the user informed that the problem is most likely on his side?	Yes	5
22	USABILITY	When no connection can be established to the authentication server, is the user informed that there is a network infrastructure problem which he cannot do anything about?	Yes	5
23	USABILITY	When no common EAP type could be negotiated, is this signalled as a configuration error to the user?	Yes	3
24	USABILITY	When the authentication server's certificate is expired, is this signalled to the user?	Yes	4
25	USABILITY	When an unauthorised authentication server (either by a mismatching CA or a mismatching server name) attempts to query the user credentials, is the user alerted about that?	Yes	6
26	USABILITY	In the manual configuration interface (if any), is the default to verify the server certificate, either via a PKI and CA chain or alternatively the certificate fingerprint?	Yes	5
27	CODE	For tunneled EAP methods, are anonymous outer identities supported?	Yes	4
28	CODE	Are there enough EAP methods supported to cover the full spectrum?	Yes	5
29	CODE	Is certificate-less mutual authentication (i.e. EAP-pwd) supported?	Yes	2
30	CODE	Can EAP-TLS be configured with an EAP-Identity different from the certificate content's CN or emailAddress fields?	Yes	3
31	CODE	Is the device tolerant regarding superfluous sending of the root CA?	Yes	2
32	CODE	Are there severe device-specific bugs?	No	5

The following seven operating systems (which fulfils KPI-2) were assessed according to these criteria:

Category	Criterion	PrivatOS 1.0.4	Blackberry OS 10.2.	Android 5.0	Mac OS X 10.10.2	MS Windows 7	MS Windows 8.1.	MS Windows Phone 8.1
CRITICAL	1	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN
CRITICAL	2	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN
CRITICAL	3	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN
CRITICAL	4	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN
CRITICAL	5	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN
CRITICAL	6	RED	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN
MAJOR	7	GREEN	GREEN	GREEN	GREEN	GREEN	GREEN	YELLOW
MAJOR	8	YELLOW	YELLOW	YELLOW	GREEN	GREEN	GREEN	YELLOW
MAJOR	9	YELLOW	YELLOW	YELLOW	GREEN	YELLOW	GREEN	YELLOW
MINOR	10	0	0	0	0	0	0	0
MINOR	11	0	0	0	0	3	3	0
MINOR	12	2	0	0	2	2	2	2
USABILITY	13	5	5	0	5	5	5	5
USABILITY	14	5	5	5	5	0	0	5
USABILITY	15	6	6	6	6	6	6	0
USABILITY	16	5	5	5	0	5	2,5	0
USABILITY	17	4	4	4	0	0	0	4
USABILITY	18	4	4	4	4	0	0	4
USABILITY	19	0	0	0	3	3	3	0
USABILITY	20	0	6	0	6	6	6	6
USABILITY	21	0	5	0	0	0	0	0
USABILITY	22	0	2,5	0	5	0	0	0

Category	Criterion	PrivatOS	Blackberry OS	Android	Mac OS X	MS Windows	MS Windows	MS Windows
USABILITY	23	0	0	0	0	0	0	0
USABILITY	24	0	0	0	4	0	0	0
USABILITY	25	0	0	0	6	0	0	6
USABILITY	26	0	0	0	5	5	5	5
CODE	27	4	0	4	4	4	4	0
CODE	28	5	5	5	5	0	5	5
CODE	29	2	0	2	0	0	0	0
CODE	30	3	3	3	3	3	3	0
CODE	31	2	2	2	2	2	2	2
CODE	32	0	5	0	5	0	0	0
SECURITY		RED	YELLOW (2)	YELLOW (2)	GREEN	YELLOW	GREEN	YELLOW (3)
SCORE		47,0	57,5	40,0	70,0	44,0	46,5	44,0

As can be seen from the assessment results, very few operating systems score highly in terms of security features. Also, most devices only achieve less than half of the potential scores in the usability/code/minor security areas.

The initial thinking for the SENSE “User-friendly and Complete EAP Supplicant” quality label was to require at least 50% (48.5 absolute) of the integer score and a GREEN for the security features.

The assessment shows that only one single supplicant would have achieved this label (Mac OS X 10.10.2). This is unfortunate and might look like subtle advertising by outside parties. To prevent this, and in order to award a label that makes a case for more than one single device, the thresholds would at least have to be lowered to “at most one YELLOW” and “absolute score of 40 points”. This would make the supplicants:

- MacOS X 10.10
- Windows 7
- Windows 8.1

achieve this label. However, the label criteria are then so low that it does not make sense to promote this label in any way.

This project task shows that the industry still needs to work on EAP supplicants a lot before the supplicant landscape can be considered user-friendly and secure. See the “Outlook” section for possible future consequences of this work; particularly a systematic follow-up to bug trackers of the supplicant vendors appears imperative to the authors.

2.1.3 EAP Lab (Task 2.3)

In this section, one of the flagship results of the project is described: the creation of the EAPlab “roaming consortium sandbox”. The section is organised in three sub-sections: first the expected results as defined in the project plan, followed by a description of the work as it was actually carried out (explaining minor derivations where they became necessary), and finally a presentation of the state of work as it was at the end of the project.

2.1.3.1 Expected Results per Project Plan

This task was created to achieve the following objective:

“To create a convenient and accessible testing environment for EAP configurator and supplicant testing with a purpose of lowering the difficulty and therefore cost of implementation and testing.”

The task’s performance should be measured against these milestones:

M2.2 RADIUS server basic functionality ready for internal project usage

M2.4 EAP Lab fully operational

2.1.3.2 Approach and realisation of results during project runtime

Idea and development

The main goal of the SENSE project as a whole was to enhance the security and ease of use of EAP based authentication. **EAP lab** was to provide a testing infrastructure that would allow testing supplicants against various possible scenarios. In the project proposal the lab was referred to as “EAP lab”, during the project this general description turned into a brand name “EAPlab”; this deliverable uses the latter name to refer to the product.

Many of these scenarios grew up from many years of NREN and organisation level operational experience of the eduroam service, where problems were observed arising from RADIUS misconfigurations of various sorts, but also simple everyday situations like a user account being disabled or a user certificate expiring. Some scenarios simulate possible man-in-the-middle attack attempts. Anyone who has some experience with configuring a RADIUS server knows that testing a device against all such situations would be extremely tedious work. In fact, SENSE was in contact with researchers whose work on security has been blocked exactly by such RADIUS configuration complexity. **EAPlab** is flexible and new scenarios can be added, as in fact has happened within the project lifetime, where new possible problems with device configurations have been uncovered.

The information and testing infrastructure has been created on an Ubuntu Linux platform. The main site: <http://supplicants.net> describes the project goal, its achievements and final results. The experimentation platform EAPlab is operated under <https://eaplab.supplicants.net>. After experimenting with several approaches for EAPlab, it has been decided to set up a single RADIUS proxy with a number of virtual RADIUS servers behind it. The RADIUS server is available in two implementations: FreeRADIUS and OSC Radiator. Each EAPlab user is provided with a unique realm (domain name) within supplicants.net. The front RADIUS proxy consults the EAPlab database and forwards packets for the given name to the backend RADIUS server responsible for a given scenario. All packet routing is done dynamically, therefore changes are applied instantaneously.

Potential user groups

EAPlab is an experimentation tool for all interested users worldwide. In particular, it has a primary use internal to the project. EAPlab was to be the evaluation platform for Task 2.2. Therefore an iteration process of implementation had been prepared, where a lab with a limited functionality would be made available internally, and as its usage grew the new functionality would be added based both on the initial plan and on experience gained. One such example can be the functionality helping users in keeping track of their tests and also of making their findings available to other EAPlab users, thus eliminating duplication of work. The need for this functionality became apparent during work within Task 2.2, but has not been thought of during the initial planning. The combination of several related tasks working in close cooperation turned out to be extremely successful.

EAPlab has been planned for and designed as an experimentation environment for many groups of users. Supplicant developers will be happy to use a RADIUS environment that is also extremely adapted to testing the resilience of their supplicants to various authentication errors resulting from RADIUS misconfiguration and possible MITM attacks. However supplicant developers are a small

group, a much wider one are site administrators whose main interest is experimenting with new devices so that they can then support their users.

For site administrators, automatic device configuration before testing should be of great assistance, and the success of eduroam CAT shows how useful the results of this work can be. Therefore EAPlab was to be equipped with its own instance of eduroam CAT. Following the “Simple & Easy” mantra of the SENSE project, the CAT configuration complexity should be reduced to the minimum. Therefore, as the user’s individual environment is created automatically, so can be the corresponding CAT configurations. This way the user is spared all work needed during manual CAT configuration, while is also given full powers of such configuration when advanced setting may be required.

One other user target group are developers of supplicants and installers that are based on the generic profile developed by SENSE. Within SENSE, Task 2.4 was developing two such generic profile consumers. The work within this task was to provide a module for the CAT software capable of producing such profiles from the CAT configuration. Such profiles should also be available through the EAPlab configurator distribution service.

Integration with CAT software

The description above clearly shows that EAPlab has a close relationship with the CAT software, therefore it has been decided to build the EAPlab interface implementation inside of the CAT source tree, as a separate subdirectory within the distribution. This approach minimised the duplication of programming work and opens the door towards future work on a more complex experimentation environment, where EAPlab tests and eduroam CAT tests can supplement each other.

EAPlab needs to provide an interface for installer download, however EAPlab has its own look and feel, adjusted to its role as a service, therefore it has been realised that the installer download interface should be developed in the same look and feel. The new user API designed within CAT version 1.1 was a perfect tool to build such an interface and for this reason EAPlab has been developed on top of the pre-release version of CAT 1.1. Also the work on generic profile delivery, which was to become the part of CAT 1.1 was an argument for this decision. An unavoidable consequence is that at the very end of the project it is necessary to move the EAPlab platform to the newest version of the CAT distribution.

For all but the most basic uses, EAPlab requires authentication, so that the working environment may be personalised for each user. It is important to stress however that no personal data is collected nor retained. During the project lifetime Google authentication was the only option for authentication, but in the last days of the project, an eduGAIN connection was realised, providing plenty of authentication sources from the academic sector.

2.1.3.3 Description of final result

Integration in overall project web presence

This screenshot shows the SENSE web site, <http://supplicants.net>. It provides information about SENSE project goals and results. EAPlab can be found at the lower-left corner.



EAP SUPPLICANT SPECIFICATION, EVALUATION, ON-LINE TESTING

SENSE

- Main page
- Project research tasks
- Standardisation of EAP Metadata File Format
- Quality Evaluation and Improvement of Supplicants
- EAP Lab – Creation and maintenance of experimentation environment
- Generic configurator development
- Project services and results
- Standardisation
- EAP lab
- Supplicant evaluation
- Generic configurator
- Contact

Introduction

SENSE is a project which aims to improve the usability of Enterprise Networks (wired and wireless) while maintaining maximum security. The primary target for the improvements created by this project is a GN3plus service: eduroam.

With this project, for the first time, a comprehensive set of criteria will be defined which together describe what it means to implement a secure and yet easily usable EAP supplicant for enterprise network usage. Further to this, a number of existing EAP supplicants will be assessed according to these criteria, which leads to a register of recommendable supplicants, and a list of caveats to look out for when either procuring devices for an organisation or when providing helpdesk support for these devices. The project even goes a step further than just contemplating the current state-of-the-art: for select supplicants, improvements to their code base will be developed which increase the usability and security of these supplicants. The targeted supplicants are: A generic Linux installer, compatible with many Linux distributions, and an Android App for Android versions 2.2 and above. The project will also create a "EAP Metadata file format" and implement it on the aforementioned platforms to create a common way of sending EAP configuration to supplicants. This will set an example for other platforms which currently use proprietary, custom, and incompatible formats.

The project contributes to the goals of the Open Calls by implementing all the listed objectives of Open Call #16. It also contributes to the overall goals of GN3plus by defining and implementing real-life improvements to the GN3plus service "eduroam" in terms of end-user usability of the service.




European Commission Communications Networks, Content and Technology



PROJECT PARTNER




PROJECT COORDINATOR

EAPlab

The main goal of the task was the preparation of the test environment for task 2.2. This goal has been fully achieved and the resulting environment is powerful, user-friendly and works flawlessly. A very good example of its usefulness is the eduroam CAT development work. eduroam CAT contains a testing environment targeted at discovering RADIUS server configuration flaws. Debugging these tests required verification against RADIUS servers having the configuration flaws that the tests were supposed to uncover. EAPlab provided exactly this opportunity and saved untold amounts of time that would have to be spent on manual configuration of such servers.

In order to make use of EAPlab, the user needs to set up a wireless access point, select WPA2 enterprise network authentication and direct the access point to the front-end RADIUS server of EAPlab. The actual testing is described below.

EAPlab is accessible at <https://eaplab.supPLICANTS.net>



EAP SUPPLICANT SPECIFICATION, EVALUATION, ON-LINE TESTING

SENSE

SENSE main page

About EAPlab

Documentation

Terms of Use

Privacy Policy

Use EAPlab

Fixed configs

Login and do a lot more ...

EAPlab provides a RADIUS environment allowing you to test EAP supplicants in various scenarios.

How can I use EAPlab?

There are several uses we can think of:

802.1X lab

Suppose you want to test 802.1X, either wireless (WPA2-Enterprise) or wired. You have the necessary networking equipment and you would like to test authentication without the hassle of setting up a RADIUS server. This could be a student assignment or a training lab. As long as your target is something like a wireless controller, you want to focus on its functions and not spend time on learning about RADIUS deployment. Here EAPlab is your friend. All you need to do, is to point your network equipment to our RADIUS server and use one of our fixed configurations. You can use the fixed configurations without any registration.

Experimentation at home

Modern wireless home devices usually support WPA2-Enterprise, so you can easily set up such a network if you have access to a RADIUS server. Since your home IP is probably changing, it is not so easy to set up a connection to any production RADIUS server unless you resort to something like VPN. EAPlab provides you with a RADIUS server which is accessible from everywhere, so your home experiments will run just fine.

New device testing

A new wireless device appears and you want to test how it will behave, what EAP types will it handle, whether you can configure it safely, etc.

The lab can simulate a correctly working RADIUS server, or at your choice can intentionally misbehave in numerous ways. That way, you can test the new device's behaviour in very many success and failure scenarios. The best approach is to set an individual EAPlab profile to configure the device, connect, then change the RADIUS server settings reconnect and observe.

Testing XML configuration profiles

In SENSE project we work on describing EAP configuration in a universal XML profile. Using the provided CAT instance you can generate various profiles and check how your supplicant will consume it. You can also test your configuration against the EAPlab RADIUS servers.

What EAPlab is not

Being an EAPlab is **not** a RADIUSlab, i.e. we put emphasis on settings important for EAP but not on various other useful RADIUS features like accounting, VLAN setting, Chargeable-User-Identity and many others. If you think that it would be great if we did support these missing functions, let us know. If enough potential users find this useful, we could decide to extend the functionality.

What you should NOT use EAPlab for

EAPlab can be used anonymously or by authenticated users. Unauthenticated usage provides basic functionality, while the optional user authentication provides a customised work environment with much more control over the RADIUS server behaviour and extended functionality. The following screenshot shows the unauthenticated basic functionality:

SENSE main page

About EAPlab

Documentation

Terms of Use

Privacy Policy

Use EAPlab

Fixed configs

Login and do a lot more ...

?

Fixed configs

While you are welcome to use the basic capabilities of EAPlab without any registration, we strongly advise you to set up an account and be able to do a lot more. You will not be asked to provide any personal data in order to set up your own configuration.

The RADIUS server is available at the address **radius.suplicants.net:1812 (150.254.191.209:1812)**. All RADIUS clients are accepted. The RADIUS secret is **sense_is_great**.

EAP methods currently supported:
FreeRadius: TTLS-PAP, TTLS-MSCHAPv2, PEAP-MSCHAPv2, EAP-PWD, TLS;
Radiator: TTLS-PAP, TTLS-MSCHAPv2, PEAP-MSCHAPv2, EAP-FAST, TLS.

Provided CAT installers will set SSID **SENSE**.

You can access three RADIUS configurations:

Single CA
*This server has been configured with a certificate provided directly from a root CA.
access with **User-Name eaplab@r1.suplicants.net**; password **eaplab**.
For EAP-TLS install the user certificate from this [P12 file](#) also protected with passphrase **eaplab**.
Server certificate name: **radius.suplicants.net**
You can download the root CA certificate as [PEM](#) or [DER](#).*
Download [CAT installer](#) for this configuration.

CA chain
*This server has been configured with a certificate provided from an intermediate CA below a root CA.
access with **User-Name eaplab@r2.suplicants.net**; password **eaplab**.
For EAP-TLS install the user certificate from this [P12 file](#) also protected with passphrase **eaplab**.
Server certificate name: **radius.suplicants.net**
You can download the root CA certificate as [PEM](#) or [DER](#) and the intermediate CA certificate as [PEM](#) or [DER](#).*
Download [CAT installer](#) for this configuration.

TCS chain
*This server has been configured with a certificate provided by the TCS service
access with **User-Name eaplab@r3.suplicants.net**; password **eaplab**.
For EAP-TLS install the user certificate from this [P12 file](#) also protected with passphrase **eaplab**.
Server certificate name: **suplicants.net**
You can download the root CA certificate as [PEM](#) or [DER](#) first intermediate CA certificate as [PEM](#) or [DER](#) second intermediate CA certificate as [PEM](#) or [DER](#).*
Download [CAT installer](#) for this configuration.

As can be seen from that screenshot, the unauthenticated basic scenario provides the “default” case: a fully working RADIUS Server which supports very many EAP methods, and a fixed set of user credentials (password-based and with client certificates) for client authentication to SENSEs RADIUS infrastructure. For the server-side credentials, the basic scenario provides three different server certificates: a certificate signed directly by a root CA; one which is signed by an intermediate CA which in turn is signed by a trusted root; and one specific chain of intermediate certificates which is particularly popular in eduroam (TERENA Certificate Service).

The true power of EAPlab is unleashed after the user creates an individual profile. No personal data is required. Authentication is based on the simpleSAMLphp package, and can be configured to support most of the current authentication methods.

Welcome **Tomasz Wolniewicz**

My profile

name (your realname or a nickname, just so that you can recognise that you are logged in)

username (just the user part, the realm will be set by the system)

password (it will be saved as clear text in the Lab database)

ssid (SSID to be configured in CAT profiles (installers will fail if this is empty))

After configuring the profile, the user obtains access to the full testing environment. The interface can be seen below.

My tests

Test device

Your selected test device is: **CAT tests**. This can be changed on **My devices** page.

You may save your individual tests using the *device comments* links next to each of the tests. Green links correspond to the ones where you have already saved complete results, orange links are ones that you started to work with, but did not mark as complete, the blue ones are the untouched ones.

Network authentication data





The RADIUS server is available at the address **radius.suppliants.net:1812 (150.254.191.209:1812)**. All RADIUS clients are accepted. The RADIUS secret is **sense_is_great**.

In order to use your customised configurations you need to authenticate to the network.

- For username/password based EAP methods authenticate as **tw1@user722.suppliants.net**.
[Show password: *****](#)
- For certificate based EAP methods (EAP-TLS) use the name corresponding to the particular certificate, as shown on the **My certificates** page.

Changes to the settings are applied instantly.

Available configurations:

Name	download CA cert	SSID	download CAT installer	modify CAT settings
<input checked="" type="radio"/> Single CA -1	PEM or DER	SENSE		
<input type="radio"/> CA chain	PEM or DER	SENSE		

Select RADIUS implementation

☒ FreeRadius 3.02 (supported methods: TTLS-PAP, TTLS-MSCHAPv2, PEAP-MSCHAPv2, EAP-PWD, TLS).

☐ OSC RADIATOR 4.13 (supported methods: TTLS-PAP, TTLS-MSCHAPv2, PEAP-MSCHAPv2, EAP-FAST, TLS).

Configuration variants:

These variants will allow you to test the supplicant behaviour when something unexpected or badly configured happens.

- ☒ default configuration ([more info](#)) ([device test comments](#))
- ☐ immediate Access-Reject ([more info](#)) ([device test comments](#))
- ☐ Access-Reject after EAP conversation (default CA) ([more info](#)) ([device test comments](#))
- ☐ No reply ([more info](#)) ([device test comments](#))
- ☐ No EAP match ([more info](#)) ([device test comments](#))
- ☐ default CA, correct name in the subject, no subjectAltName ([more info](#)) ([device test comments](#))
- ☐ default CA, different name in the subject, no subjectAltName ([more info](#)) ([device test comments](#))
- ☐ default CA, different name in the subject, correct name in subjectAltName ([more info](#)) ([device test comments](#))

When the user points an access point (or other IEEE 802.1X device) to the EAPLab RADIUS server then connection attempts of the realm assigned to this user are routed to the selected scenario. In the control panel the user can select a number of settings:

- Type of server certificate. RADIUS servers can use (a) a local CA which normally issues the certificates directly, (b) a set of local CAs with intermediates, or as is the case in many deployments, for instance also in eduroam, (c) general purpose certificates from well-known CAs are also used. Such certificates are never issued by the corresponding root CA, but by a subordinate intermediate CA. It has been observed that some devices may have a configuration problem when a certificate chain is present, so EAPLab allows to test for all these cases.
- RADIUS implementation. EAPLab provides FreeRADIUS and OSC Radiator. This may help in some very subtle cases when a user may want to be absolutely sure that a problem is not RADIUS implementation specific. Radiator can also be used for testing of the EAP-FAST method which is not well implemented in FreeRADIUS.
- Configuration variants. At the moment of this writing there were 18 scenarios available. Each of the scenarios has a *more info* popup, which describes the purpose of a given test, expected connection result, and also allows to show the details of the RADIUS server certificate. It must be understood that the crucial element of wireless connection security is the RADIUS server certificate. The supplicant in a user's device should be able to identify the server based on the data in the certificate and either accept or reject the connection, **before** sending any sensitive information, like the user's password. The scenario called ***certificate from another CA*** emulates such a MITM attack, where an attacker sets up a rogue network authenticated by a local (also rogue) RADIUS server. The attacker normally is unable to obtain a server certificate from the CA that the user has authorised, therefore even if all other details like the server name in the certificate are as expected, the difference in the issuing CA should raise the alarm. If a supplicant does connect then it means that the given device (or a given configuration) is a real hazard for its users.

X
Certificate from another CA

This configuration differs from the default one by the root CA in the server verification path. Server certificate has all extensions which may be required by known supplicants.

Expected connection result **FAIL**

Download CA cert as [PEM](#) or [DER](#)

[show server cert details](#)

Subject:
CN=radius.suplicants.net,OU=SENSE,O=GEANT,C=EU

Issuer:
CN=EAPLab Root CA 1,OU=SENSE,O=GEANT,C=EU

Valid from:
Friday, 31-Oct-2014 11:41:08 GMT

Valid to:
Thursday, 27-Jul-2017 11:41:08 GMT

Serial number:
4096 (0x1000)

SHA1 fingerprint:
1b632f364375705e5cae2820f760153d27a64bc4

Extensions:
extendedKeyUsage: TLS Web Client Authentication, TLS Web Server Authentication
crlDistributionPoints: Full Name: URI:http://www.suplicants.net/sense1.crl
basicConstraints: CA:FALSE
keyUsage: Digital Signature, Key Encipherment
subjectAltName: DNS:radius.suplicants.net

download server cert: [PEM](#) or [DER](#)

After performing the test the user can examine the server side results (the contents of the window can also be downloaded to the local disk).



My tests

✕ 'user346.suplicants.net'

EAP-Type
TTLS

Inner EAP

Packet-Type
Access-Request

User-Name
'twoIn@user346.suplicants.net'

Realm
'user346.suplicants.net'

Realm
'user346.suplicants.net'

Inner EAP

Packet-Type
Access-Accept

2015-03-13 12:38:02

Packet-Type
Access-Accept

MS-MPPE-Recv-Key
0xb44dfdc578777c28708f57ed7ae2198f298d350b53681e16145b294d37b86183

MS-MPPE-Send-Key
0x62b53ce953cb7292f9d4fc40963c6a387dcb7fd6a520db226e1380f70401106b

EAP-MSK
0xb44dfdc578777c28708f57ed7ae2198f298d350b53681e16145b294d37b8618362b5
3ce953cb7292f9d4fc40963c6a387dcb7fd6a520db226e1380f70401106b

EAP-EMSK
0x175288af368967954b81af6d3bc631ec192b95ac876f4debb310afd13fa80e12f5fa
e63610b8da690da8b8a13fb8c7c01cb10dbcf6e75a0760a233e82bd9bab4

EAP-Session-Id
0x15556a6a2c6cee9a98afc1e5a176d91dc719d5617e619e64a80e70860d799fa60608
ad3f37dbb83c545598404c026ad0fbc3dbd5ad11b7114633b84655451e90b3

EAP-Message
0x03040004

Message-Authenticator
0x00000000000000000000000000000000

User-Name
'twoIn@user346.suplicants.net'

Proxy-State
0x34

☐ Access-Reject after EAP conversation (default CA) ([more info](#)) ([device test comments](#))

☐ No reply ([more info](#)) ([device test comments](#))

☐ No EAP match ([more info](#)) ([device test comments](#))

For a more thorough examination, a full debug log from the server is available for download. An extract of such a log is presented below.

```
rad_recv: Access-Request packet from host 150.254.191.209 port 60791, id=54,
length=1717
```

```
User-Name = 'twoIn@user346.suplicants.net'
NAS-IP-Address = 127.0.0.1
Calling-Station-Id = '22-44-66-CA-20-00'
Framed-MTU = 1400
NAS-Port-Type = Wireless-802.11
```

[illegible]

Each scenario also provides a link to a utility in which the user can record the results of the test, mark what has been tested, give a quality mark and also mark it in the test notes as complete. The control panel clearly marks scenarios which have been completed, partially assessed or not touched at all. During the device testing that was done in task 2.2 it was realised that keeping track of so many possibilities without assistance from the system is nearly impossible.

Device	CAT tests PEAP
Test	default configuration - Configuration matching default settings. These settings are reflected in the CAT profiles. Server certificate has all extensions which may be required by known supplicants. Server certificate subject CN is radius.suppliants.net. SubjectAltName is set to DNS:radius.suppliants.net.
Expected connection result	PASS
RADIUS result	<button>get result</button>
Obtained connection result	<input type="radio"/> FAIL <input checked="" type="radio"/> PASS
Tested EAP types	<input type="checkbox"/> FAST-GTC <input checked="" type="checkbox"/> PEAP-MSCHAPV2 <input type="checkbox"/> EAP-pwd <input type="checkbox"/> TLS <input type="checkbox"/> TTLS-GTC <input type="checkbox"/> TTLS-MSCHAPV2 <input type="checkbox"/> TTLS-PAP
Comments	<div></div>
Supplicant quality	<input type="radio"/> 0 <input type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4 <input type="radio"/> 5 <input type="radio"/> 6 <input type="radio"/> 7 <input type="radio"/> 8 <input type="radio"/> 9 <input checked="" type="radio"/> 10
Last description change	2015-02-20 23:21:20
Test complete	<input checked="" type="checkbox"/>

Next test

Device tests can be made public, i.e. available to other EAPlab users.

Public devices

iOS 7 - manual config	your device
iOS7 - CAT setup - PEAP	your device
iOS 8.0.2 - TLS - CAT config	your device
iOS 8 - username/password	your device
Windows 10	your device
EAPOL TEST	
NEXUS10-gareth	
NEXUS5-gareth	
Blackberry Z30	
CAT tests	

Each of the devices on the list can be expanded to show summaries of all tests ...

Public devices

iOS 7 - manual config	your device	
iOS7 - CAT setup - PEAP	your device	
Test name	quality	complete
default configuration	10	✓
immediate Access-Reject	10	✓
Access-Reject after EAP conversation (default CA)	10	✓
No reply	10	✓
No EAP match	10	✓
default CA, correct name in the subject, no subjectAltName	10	✓
default CA, different name in the subject, no subjectAltName	8	✓
default CA, different name in the subject, correct name in subjectAltName	10	✓
default CA, correct name in the subject, different name in subjectAltName	0	✓
default CA, correct name in subjectAltName, subject empty	0	✓
default CA, different name in subjectAltName, subject empty	0	✓
default CA, certificate without the CRL pointer	10	✓
default CA, no CA:FALSE in server certificate extensions	10	✓
default CA, certificate listed on the CRL	10	✓
default CA, expired certificate	8	✓
Server sending root CA cert	10	✓
Server cert signed with SHA-1		untested
certificate from another CA	8	✓
iOS 8.0.2 - TLS - CAT config	your device	

... and even further expanded to show each individual test:

Device test details

Device	iOS7 - CAT setup - PEAP - iPad 3 CAT profile downloaded from EAPlab		
Test	default CA, correct name in the subject, different name in subjectAltName - The CN value in the subject of the server certificate is correct, but the subjectAltName contains one name and it is different from the expected server name. If the supplicants consults both the subject and subjectAltName it should pass.		
Expected connection result	PASS		
Obtained connection result	FAIL		
Tested EAP types	FAST-GTC TLS TTLS-MSCHAPV2	✓ PEAP-MSCHAPV2 TTLS-GTC TTLS-PAP	EAP-pwd
Comments	Immediate reject, message 'unable to join the network'		
Supplicant quality	0		
Last description change	2014-09-26 14:21:58		
Test complete	YES		

Previous test Next test

default CA, expired certificate	8	✓
Server sending root CA cert	10	✓
Server cert signed with SHA-1		untested

EAPlab also supports personal certificate based authentications (i.e. EAP-TLS). Such tests require a certificate distribution service, which has been implemented. Three types of user certificates can be issued, emulating the most common use cases.

TLS user certificates

TLS testing can be quite complex due to many possible settings of the user certificate. This is why we provide a few cases illustrating the most common situations. Some supplicants will behave differently depending on the number of certificates they can pick from, therefore it is reasonable to test with a single and many certificates installed.

Supplicants tend to select the RADIUS user-name from the certificate. In most cases the CN value from the subject will be used, but sometimes this setting will be inappropriate.

This page provisions TLS user certificates, which are protected with a password selected independently of the password for other EAP methods.

Certificates are issued by a dedicated CA in the form of PFX (P12) files containing user certificate, private key and the issuing CA certificate. The files will be protected with the password you set during the certificate generation.

Generate your TLS certificate

Type of certificate
☒ certificate subject CN equal to the EAP username
☐ certificate subject CN not equal to the EAP username
☐ EAP username in the certificate email, while CN not equal to the EAP username

CN data for this certificate

Passphrase for the P12 file

Username for authentication:
aa@user346.suplicants.net
Your certificate subject:
C=EU, O=SENSE, CN=aa@user346.suplicants.net

Download your TLS certificate

If you have previously generated any certificates you can download them by clicking on one of the links below:

Certificate type	Create time	Validity	Username	PFX password
CN_eq_username	2015-03-13 13:27:55	365 days	test1@user346.suplicants.net	1234

Each page of EAPlab has a help popup, extensive documentation is also provided under the "Documentation" link.



Fixed configurations

This is the simplest way to use EAPLab, but also one which does not provide many testing scenarios. You can access our preconfigured RADIUS servers with one username which is common to all. This is mainly useful if you want to test a new device or supplicant software and make sure that it will properly connect when the RADIUS server work correctly. These configurations will **not** allow you to test how the device will behave when something unexpected happens, like when the server certificate sent by the network does not match the one you have configured on your device. To test such situations you need more customisation, which is provided by other areas of the LAB.

Together with these RADIUS configurations you can also test CAT installers, in particular the generic XML profiles. The servers support a number of EAP methods. The normal CAT setup would be to pick the best EAP method for a given device, to make sure that a maximum variety of devices are supported. This is the best approach to support users, but can be misleading when testing, therefore for EAPLab we have selected another approach. A number of configuration sub-profiles are defined, each supporting just one EAP method. This way you will be sure which EAP method gets configured on your device, or an installer will not be available if a given EAP method is not supported.

are accepted. The RADIUS secret is **sense_is_great**.

EAP methods currently supported:

FreeRadius: TTLS-PAP, TTLS-MSCHAPv2, PEAP-MSCHAPv2, EAP-PWD, TLS;

Radiator: TTLS-PAP, TTLS-MSCHAPv2, PEAP-MSCHAPv2, EAP-FAST, TLS.

Provided CAT installers will set SSID **SENSE**.

You can access three RADIUS configurations:

Single CA

*This server has been configured with a certificate provided directly form a root CA. access with **User-Name eaplab@r1.suplicants.net**; password **eaplab**. For EAP-TLS install the user certificate from this [P12 file](#) also protected with passphrase **eaplab**. Server certificate name: **radius.suplicants.net** You can download the root CA certificate as [PEM](#) or [DER](#).*

Download [CAT installer](#) for this configuration.

EAPlab internals

As it has been already mentioned, upon registration, each EAPlab user obtains a unique realm within **suplicants.net**. The realms are in the form of **userXXX.suplicants.net**, where XXX is a number generated during user registration. The realm is recoded in the EAPlab database, together with other elements of the user's profile. EAPlab database stores users' profiles, RADIUS credentials, test comments, and scenario settings.

Each scenario has its version for both implementations (FreeRADIUS and Radiator) and for each of the two types of server certificate (one issued by self-signed CA and another one issued by an intermediate CA within a two-CA chain).

To the outside, EAPlab provides a single RADIUS interface to which all authentication requests (RADIUS Access-Request packets) need to be sent. This front-end RADIUS server acts as a proxy and is responsible for delivering the requests to the appropriate target – a virtual RADIUS server responsible for a single scenario. Upon receipt of an Access-Request packet, the front-end server consults the EAPlab database and locates the realm found in the packet. If the realm is found, then it the user settings (most importantly the selected scenario) can be retrieved. The front-end server forwards packets to the backend RADIUS server responsible for the selected scenario. Packets that do not match any registered user realms are rejected. A change in the database immediately changes packet routing, thus a change of a scenario can be done with a single mouse click.

The backend servers have been set up on two RADIUS implementations and the actual approach to configuration depends on the particular implementation. In the case of FreeRADIUS (version 3) each scenario has two dedicated virtual servers listening on specified ports - one for the outer connection

(from the front-end proxy) and one for the inner EAP. Each virtual server uses an EAP module appropriate for a given scenario. In the case of OSC Radiator, the server handles all scenarios on one authentication port. The Radiator realm configuration allows placing all parameters required to route packets through an appropriate scenario.

Currently, all combinations of scenarios, implementations and certificate types bring the count of possibilities up to 72. Managing such a set of tests needs to be automated both to save work and avoid errors. A set of shell and Python scripts were provided to make it easy to create the configuration for a new scenario or to do some modification to an existing one.

In order to implement all current scenarios a number of certification authorities had to be created which was achieved using shell scripts running openssl commands with dedicated openssl configurations.

RADIUS logs

One of the requirements of the project was to enable detailed log viewing. This turned out to be quite a complex task since many users may be sending packets to the server in the same time and the logs shown to a particular user should reflect only the history of this user. While at the first glance it might appear to be simple since all requests are clearly marked with the User-Name attribute containing the unique user realm, in reality the complexity stems from the fact that responses and packets from inner EAP are not identified as easily. Another challenge was to provide a similarly looking output, regardless of the server implementation. The debug output from both servers had to be provided as well, which required per user filtering of the outputs.

EAP-TLS certificates

EAPlab contains a module responsible for user certificates as used in EAP-TLS. In real life, user certificates may come in various flavours. EAP supplicants usually work best with certificates whose subject contains a CN value with the user identity required for RADIUS routing (e.g. CN=johndoe@user123.suplicants.net). Many supplicants select this value as a pre-set in their user interface or even hardwire this value as the User-Name, omitting the username setting from the user altogether. However, general-purpose user certificates are not necessarily built this way (e.g. contain an actual name like CN=John Doe) and in order to achieve an authentication, the user needs to set the EAP username manually. Typical commercial user certificates contain additional identity information besides the CN: there is e-mail information in the subject, which in many cases may coincide with the EAP identity to be used. While SENSE participants are not aware of any supplicants making use of this information, it would appear a natural thing to do, therefore EAPlab also provides user certificates which imitate this situation so that EAPlab users can verify supplicant behaviour also in these cases. EAPlab is also capable of producing short-term (24h) user certificates, this allows testing how supplicants react to expired user certificates.

Generic CAT module

The generic eduroam CAT module has been prepared as proposed and is now available for testing as a part of CAT 1.1 beta distribution at <https://cat-test.eduroam.org> as the **eap-config** device. The module delivers XML files described by the profile definition in the version 00. Within the EAPlab CAT instance the module provides a number of generic devices serving either a single EAP method or a full

set (to allow selection on the client side). The module has been extensively used (therefore also tested) by Linux and Android implementers (SENSE Task 2.4). The motivation to rely on the 00 version of the internet draft has been already described, but it must be mentioned that the implementation of the module is quite flexible and updating it to any new definition will be quite straightforward and can be done in a short time frame.

2.1.4 Generic Configurator Development (Task 2.4)

This subsection describes two supplicant implementations that were carried out as proof-of-concept work for the EAP metadata configuration file format. The section is organised in four sub-sections: first the expected results as defined in the project plan, followed by a description of the work as it was actually carried out, and finally for both implementations an overview of the implementation work, design choices and a description of the final implementation.

2.1.4.1 *Expected Results per Project Plan*

This task has the following objectives:

"To implement the generic profile definition and universal 802.1x configurators"

Measurement of success of this task is via the following Key Performance Indicator:

KPI-4 The project has created at least two implementations of a supplicant front-end which can consume the EAP metadata format

2.1.4.2 *Approach and realisation of results during project runtime*

Two supplicants have been selected for implementation – Android and Linux (for the NetworkManager supplicant backend). In the case of Android an implementation of a universal application, which can then consume a configuration profile is a necessity. Android can only be configured with an additional application, but the approach which has been used successfully for Windows i.e. preparation of a custom application, cannot be used due to a specific model of application distribution via the Google Play Store. With the proliferation of app stores for many platforms, this model may become a norm in many cases (e.g. the Windows Store in recent versions of Microsoft Windows), therefore the approach taken within SENSE using one application with separate configuration files may become much more widespread.

Both implementations have been completed and successfully tested.

2.1.4.3 *Description of final result - Linux*

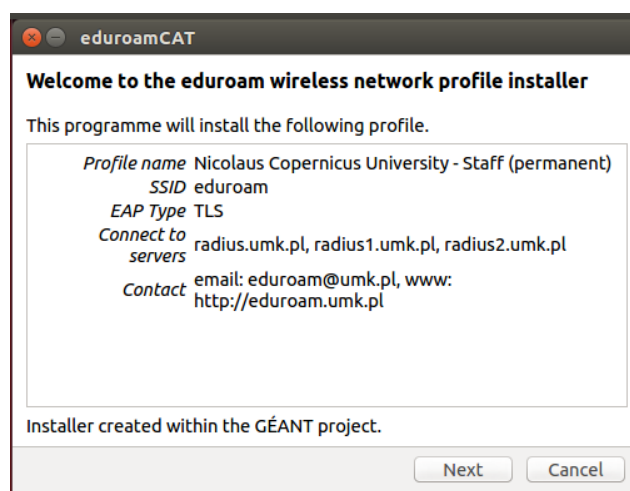
The approach to network profile installation is based on working with NetworkManager - the most popular network management software for Linux systems. NetworkManager stores configuration in

D-Bus - a system for programme intercommunication. D-Bus became widely used and is suitable for interacting with many system services. The NetworkManager daemon manages network interfaces on the basis of stored configuration. The traditional way of entering network configuration is done by using NetworkManager applets. The most common applets are: nm-applet for GNOME desktop environment and Plasma NM (formerly KNetworkManager) for KDE. Applets give access to the most of the settings, but the number of options might be too overwhelming for beginners. For this reason the developed installer replaces the function of applets.

The installer reads XML files (eap-config) which contain the network configuration and stores the configuration information in NetworkManager using its D-Bus interface. Configurations added with the installer can be seen and inspected verified in applets but some options may not be visible in those user interfaces. The installer asks a user only for credentials such as username, password or personal certificate.

The installer is implemented in the programming language Python. It interfaces with the system via the system D-Bus library and the GUI is based on the QT4 library. The SENSE installer reuses parts of an earlier code of the eduroam CAT Linux installer, developed by the same author: the SENSE implementation reuses the core (i.e. D-Bus interface) but adds a more advanced GUI and most importantly the eap-config configuration file processing.

The standard way to install the developed installer is the same as for other typical Python modules (through setuptools). The installer can be downloaded from the project website. The following is a screenshot of the installer while being executed:



2.1.4.4 Description of final result – Android

Background

Android has become one of the most popular operation systems in use for mobile devices, with a 76% market share, reaching 1 billion devices shipped carrying the OS by the end of 2014 [1]. Since Android 4.3 release, there has been support for enterprise wireless network configuration using the SDK. Prior to 4.3 there is no official support to securely configure enterprise wireless networks using the SDK. As of March 2015, 49% of Android devices in use now support these new features. [2]

With the additions in version 4.3 onwards comes support to fully configure an enterprise wireless network, including installing certificates, setting EAP methods, outer identities and setting the subject match on CA certificates. Android natively supports the EAP methods of EAP TLS, PEAP, TTLS and PWD. It also supports phase 2 methods of PAP, MSCHAP, MSCHAPv2 and GTC.

Unfortunately, the native interface to configure wireless networks does not expose some of these settings to users, and can vary depending upon vendor modifications of Android. As a result, most if not all users of eduroam who have manually configured their connection have not done so fully, but only set the minimum requirement to connect. This is sufficient to bring them online, but does not provide protection for them: they do not properly verify that they are connecting to the genuine network and are thus susceptible to attacks from so-called “rogue access points” or “evil twin networks”.

Solution overview

The app solves this challenge by making users aware of any misconfiguration or missing essential settings for their eduroam configuration, and providing the functionality to configure their devices fully and securely, while maintaining ease of use.

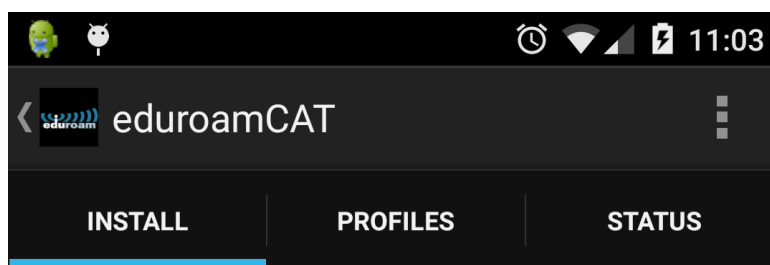
As Android devices need apps to be published through Google’s app repository, the “Play Store”, there can only be one instance of the app. If there was a separate application for each institution, the Play Store would become a very confusing place for users with thousands of “eduroam XYZ” apps and become unmanageable for administrators and developers. With having one instance of the app comes the requirement of the means to pass configuration and customisation information to the app. The eap-config file format is the solution to this requirement, and contains all the information needed to customise the app to become useful to a user.

The eduroam CAT app will be automatically launched once an .eap-config file is selected on an Android device, where the app has previously been installed. The eap-config file can be distributed by web links, in email attachments or by passing the file onto the storage of a device.

Once an eap-config file is installed by a user, the app becomes primed with the necessary information to install a secure eduroam connection. The app then takes any credentials from the user before then installing an eduroam wireless profile, and attempting to connect the user to eduroam if it is in range.

While the eap-config file can contain more than one authentication method, the app will attempt to install each method in the order they appear in the eap-config, and once one is successfully installed it will stop.

The eduroam CAT app also has a 'Status' section to provide real time feedback to users on the status of their wireless connection. It also has a debug option, to allow analysis of the supplicant and network state in Android. There is also an 'Advanced Mode' that allows a user to manually choose which authentication method to use.



Current device configuration:

- ✓ Found SSID "eduroam" with CCMP/TKIP
- ✓ Anon ID=anonymous@swansea.ac.uk
- ✓ User ID=testuser@swansea.ac.uk
- ✓ EAP Method=PEAP with Phase2:MSCHAPv2
- ✓ CA Certificate OK
- ✓ Server Subject Match=.swan.ac.uk

Username: testuser@swansea.ac.uk

Password: _____

**Installing a profile will replace any
existing eduroam settings**



An "Identity Provider Administrator Manual" was written by the author of the app, which also includes further screenshots of the application. It can be found in Appendix A.

The Android eduroam CAT app has been published in the Google Play store in March 2015. It can be downloaded from the Play Store app (direct link: <https://play.google.com/store/apps/details?id=uk.ac.swansea.eduroamcat>) and it is also available from the SENSE web space (<https://www.suplicants.net>) .

Future work to enhance the eduroam CAT app includes the ability to automatically discover eap-config profiles using a discovery process, or functionality from the cat.eduroam.org website. There is also room to improve the TLS experience as bugs are fixed, and hopefully functionality added. There will inevitably be on-going work to fix bugs with the existing app, as well as implementing any possible feature requests and suggestions from the community.

Implementation challenges

Android requires credential storage to be enabled for enterprise wireless network profiles to be saved, so a user of the app must have this enabled. In version 4.3 credential storage can only be enabled when the user sets up his own storage password. This requirement has proven to be a very troublesome and confusing feature for users – but it is unavoidable when EAP methods using a server certificate and corresponding Certification Authority (CA) are to be configured.

In version 4.4 and higher, credential storage is enabled and secured using the screen lock settings. This includes passwords, pins, and combination swipes. Without credential storage / screen lock enabled, the app cannot function. Unfortunately, there is not way around this, but the eduroam CAT app checks and prompts users to enabled screen lock if it is not enabled yet.

While the app can successfully configure and connect PEAP and TTLS profiles, TLS has proven to be difficult as a result of some bugs with the Android SDK. With version 4.3 and 4.4 installing a TLS connection fails if the private certificate is not included with the profile. There is no mechanism to select a user-defined certificate that is already installed in the device [3]. With version 5.0 and higher, the app can install the TLS profile successfully, and let the user manually select which personal certificate to use during the first connection. This still requires manual intervention by the user [3]; an automated API call to search the certificate store and link the personal certificate automatically would be more user-friendly. EAP-TLS still needs more testing at this point, with an official bug to be reported to Google.

Another bug discovered was the `getCaCertificate()` method, which always returns null on a installed profile. This made testing for the fact whether the CA certificate is installed difficult. A workaround was found – by serialising the whole enterprise profile, and testing for a given string. This is an ugly approach, and needs to be ameliorated once the bug is fixed. [4]

Using self-signed certificates in Android also results in a confusing message to users, stating that the network may be monitored by an unknown third-party. If the CA certificate is installed into the “Wi-Fi” store, it is only usable for EAP authentication purposes and the message is factually incorrect. This is a known issue in Android, which will hopefully be fixed or improved. A corresponding bug report is open at [5].

The eduroam CAT app user interface design and implementation proved difficult, given the many combinations of android versions, vendor customisations, device types, screen sizes and orientations. Development and testing of the app UI on one set of devices often resulted in different rendering on other devices. Even testing the UI in the many emulated device types available through the SDK proved ineffective. This resulted in keeping the UI as simple as possible in order to limit the number of devices that may render the UI unexpectedly.

The SENSE app

The SENSE app is a functional copy of the eduroam CAT app, but with the SENSE SSID set instead of eduroam. The SENSE app can be used to perform testing/comparison of Android's wpa_supplicant in conjunction with the EAPLab.

The SENSE app would ideally also have the ability to parse a WirelessConfiguration profile, similar of the eap-config, but containing wireless only information such as SSID. This would allow the app to expand beyond being a SENSE only testing app, to a generic enterprise wireless testing app.

[1] <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>

[2] <https://developer.android.com/about/dashboards/index.html>

[3] <https://groups.google.com/forum/#!topic/android-developers/xEvkLI5Aqvs>

[4] <https://code.google.com/p/android/issues/detail?id=160819>

[5] <https://code.google.com/p/android/issues/detail?id=82036>

2.2 Management and Dissemination (Work Package 1)

This work package contains two elements: project management and dissemination of results. The following two sections describe the work undertaken for these two tasks.

2.2.1 Project Management (Task 1.1)

As per the project plan, this task should deliver the following:

“Projects will be managed using GÉANT's Project Management Framework (PMF). This task will provide procedures that are needed for project coordination. It will involve:

- *coordination with relevant Activity Leader, and where appropriate the Open Calls Co-Ordinator throughout the year with monthly and quarterly reports.*
- *It will also ensure internal coordination and communication within the project.”*

The following deliverable relates to this task:

D1.1 Report on the Project Results

The project was managed in a standard way according to the PMF.

In the beginning of the project, monthly meetings with the Technical Coordinator were held. The frequency of these meetings was subsequently lowered to quarterly meetings. The project also delivered monthly “Red/Amber/Green” (RAG) reports to match actual progress against the project

plan.

The deliverable D1.1 was delivered on the 31st of March 2015, in line with the project plan.

2.2.2 Dissemination of Project Results (Task 1.2)

2.2.2.1 Expected Results per Project Plan

According to the project plan, the dissemination activity in the project should deliver:

The project will present its ongoing work and results at various venues throughout the project lifetime. In addition to these events, a permanent project website will be established. The following three events are targeted for dissemination:

89th INTERNET ENGINEERING TASK FORCE Meeting, London, March 2014
[...]

91st INTERNET ENGINEERING TASK FORCE Meeting, USA, November 2014
[...]

TERENA NETWORKING CONFERENCE 2014
[...]

TERENA Task Force Meeting presentation
[...]"

The following milestones relate to this task:

- M1.1 Presentation of initial draft version of EAP Metadata File Format at 89th INTERNET Engineering Task Force
- M1.2 Demonstration on TNC'14
- M1.3 Presentation of stabilised draft version of EAP Metadata File Format at 91st INTERNET Engineering Task Force
- M1.4 Demonstration on TERENA TF-Mobility

2.2.2.2 Actual Dissemination Results

The dissemination activities in SENSE achieved all of the milestones above, with a slight deviation in timing for M1.4 due to re-scheduling of the GÉANT Association's 34th Task Force Mobility and Network Middleware meeting (see below). The project also disseminated or will disseminate its results at four additional opportunities, effectively delivering twice as many outputs as targeted.

The project's results have been published on the project-owned website and domain <https://www.suplicants.net>. The SENSE team is committed to keep operating this website and to add new results on a best-effort basis even after the formal end of the project.

In addition to the project website, the following dissemination activities were executed/have been prepared for execution after project end, in chronological order:

32nd Task Force Mobility and Network Middleware Meeting, Zurich, Switzerland (February 2014)

Only a few months after start of the project, a first presentation of the content of the SENSE project and the first preliminary findings was given. The presentation was followed by a good amount of discussion. The presentation raised the community's awareness of the problem space and also provided valuable input to the SENSE work. The presentation is archived at the GÉANT Association's website under <https://www.terena.org/activities/tf-mobility/meetings/32/SENSE-TF-MNM.pdf> . The meeting minutes can be found at <https://www.terena.org/activities/tf-mobility/meetings/32/32tfmnm-minutes.pdf> .

This dissemination activity was not contained in the project plan and thus makes the dissemination activities exceed their original target. It was delivered on a voluntary basis outside of SENSE budget.

89th IETF Meeting, London, United Kingdom (March 2014)

The first version of the EAP metadata specification was presented at the 89th meeting of the Internet Engineering Task Force in London, UK. The specification was uploaded to the IETF as an Individual Submission (meaning work by an individual which is not yet endorsed by the IETF). The presentation is archived at the IETF document repository under <http://www.ietf.org/proceedings/89/slides/slides-89-opsawg-1.pdf> ; audio recordings can be accessed from the IETF web site.

This activity was part of the project plan; the attendance and presentation at this meeting completed M1.1.

TERENA Networking Conference 2014 (May 2014)

SENSE personnel attended the TERENA Conference 2014 as per project plan. An extended abstract was submitted and accepted, and a demo booth was requested and granted to show a live demo of an early version of the Android app (the demo was shown twice on two consecutive days of the conference). The timing of events was:

- Demonstration: "EAPLab – The Roaming Consortium Sandbox"
Tuesday, 20 May 2014
- Presentation: "SENSE – Secure Enterprise Networks Simple & Easy"
<https://tnc2014.terena.org/core/presentation/29>
Wednesday, 21 May 2014
- Demonstration: "EAPLab – The Roaming Consortium Sandbox"
Thursday, 22 May 2014

This activity was part of the project plan; the attendance and presentation at this meeting completed M1.2.

91st IETF Meeting, Honolulu, HI, United States of America (November 2014)

In the time following the IETF89 meeting, SENSE has worked on a revised edition of the EAP metadata specification. This new version was presented at the IETF91 meeting as planned in the original project plan. The presentation which was delivered at the meeting can be found at <http://www.ietf.org/proceedings/91/slides/slides-91-opsawg-2.pdf>, audio recordings can be accessed from the IETF web site.

Publication: CONNECT #18 Special Edition (February 2015)

Near-final results of the SENSE work were submitted for publication to a special edition of the "CONNECT" magazine in December 2014. Publication was accepted, and a two-page article about SENSE was published in February 2015 ("CONNECT – The magazine from the GÉANT community – Issue 18 – 2015"). CONNECT has significant outreach in the NREN community and in the funding agencies.

The magazine can be downloaded from

http://www.geant.net/MediaCentreEvents/news/Pages/CONNECT_issue_18_is_now_available.aspx.

This dissemination effort was not accounted for in the original project plan and thus makes the dissemination activities exceed their original target. The submission was prepared within the SENSE budget.

GÉANT Symposium 2015 and eduroam Europe SG Meeting, Athens, Greece (February 2015)

The project was given a presentation slot during the opening plenary of the GÉANT Symposium 2015 to present one of the SENSE results: the Android application. Being in a plenary presentation means significant exposure; SENSE believes that this event generated significant impact because staff members from many NRENs were present and learned about the future possibility to configure Android correctly.

Another presentation slot was allocated for a more complete overview of all SENSE results; a presentation to that end was delivered in the "Trust & Identity" track of the symposium.

As a side-event after the official Symposium, at the same venue, an eduroam Europe Steering Group meeting was held. At this meeting, a comprehensive overview of SENSE work was given to participants (some of which were unable to attend to the preceding main event), and much more detail of the particular result EAPLab was given, including a video demonstration of the EAPLab product.

Presentations of the GÉANT Symposium 2015 can be found on the GÉANT intranet; the presentation and videos which were presented at the eduroam Europe SG meeting are hosted at http://www.umk.pl/~twoIn/Athens_SENSE.zip.

These dissemination efforts were not required as per the project plan and can be considered an extra result exceeding the original expectations. They were prepared and delivered within the SENSE person month budget.

34th Task Force Mobility and Network Middleware Meeting, VC (April 2015, planned)

As per the project plan, SENSE should publicly report its results *“at the TF-Mobility and Network Middleware (or successor task force) which is closest to project end (the exact timing of meetings is not known yet)”*.

A TF-MNM meeting took place end of November 2014, which was rather early for these purposes, as the project still had four months to go with many results pending. It was envisaged to have the next meeting around the month of March 2015, which would have been perfect timing to present final results. SENSE decided to wait for this next meeting.

Unfortunately, the exact date of the meeting was fixed to 16 April 2015; a mere two weeks derivation from schedule, but this meant that the milestone M1.4 would not be delivered in M18 but in the next month; which is after the project end. The meeting will be a “virtual” meeting (VC only).

This is a minor derivation. SENSE has prepared the presentations to be given inside the project lifetime and using SENSE person months. The actual presentation will not need any budget as the participation in the VC does not incur a cost; presentations will be delivered by SENSE staff inside their normal work time, and will not be compensated for by SENSE.

M1.4 should be considered as being completed in “M19”.

TNC15 Networking Conference, Porto, Portugal (June 2015, planned)

The TNC2015 Conference is outside the SENSE project lifetime and has consequently not been planned for inside the project proposal.

However, SENSE personnel believes that this event is a major opportunity to present the actual final results of the project to a wide international audience, and that presenting at this venue will create significant impact for the project, even if in a “post-mortem” phase.

Therefore, SENSE participants have created extended abstracts for two elements of the project results while the call for papers was open. Both of these extended abstracts have been accepted, allowing for dissemination at the conference in June 2015. The two accepted talks are “Supporting user privacy, security and ease of use in eduroam” (see <https://tnc15.terena.org/core/presentation/98>) and “Easy 802.1X Onboarding with EAPConfig files and SCAD” (see <https://tnc15.terena.org/core/presentation/119>).

These dissemination efforts were not required as per the project plan and can be considered an extra result exceeding the original expectations. They were partially prepared within the SENSE person month budget; final preparations for the presentation slide decks and the actual delivery will be delivered with participants’ individual budgets outside of SENSE.

3 Project Impact

SENSE has achieved impacts in all the originally envisaged areas as per section B.5 of the project proposal.

The developed supplicant metrics serve as handbook for supplicant developers. SENSE personnel have already been made aware by a third party (ARNES) that the metrics are already actively used as a reference document for ARNES' newly-developed supplicant "ArnesLink".

EAPlab has already proven extremely valuable beyond the SENSE project borders: it is used

- to improve of the RADIUS conformance checks of eduroam CAT v1.1
- the supplicant implementation ArnesLink uses it to verify proper operation of the supplicant
- bug reports against Apple iOS have been submitted, and the bugs could easily be verified by Apple staff using EAPlab's features
- some of the EAPlab test cases were included into a third-party conformance test suite (wpa_supplicant / Wi-Fi Alliance).

eduroam Service Providers have made use of EAPlab to test their local setup against it (SP-only deployments without actual user accounts to test with). A research group from the University of Luxembourg who are concerned with system security (spanning from end user experience to the actual protocols and cryptography in systems) have evaluated EAPlab and, due to the ease of generation of test cases with it, will likely use it to conduct user or device studies.

The Android App has generated a lot of early feedback and demand from a large prospect customer base (eduroam NROs and eduroam Identity Provider administrators). It is now publicly available on the Google Play Store, with the prospect of making millions of eduroam users' life easier in the future.

The standardisation task raised IETF awareness of the existing void space regarding EAP configuration, and is on its way to publication as an RFC.

4 Challenges

Despite being a very successful project which has achieved all its expected results, a few areas were uncovered where work could not be performed as planned, or where workarounds were necessary because an issue turned out to be harder to solve than anticipated.

In particular, there were two such issues in the project. They are reported here to inform readers for possible future project planning purposes; these areas would need to be given significantly more energy or treated from a different angle to maximise impact in a future project.

4.1 KDE User Interface Improvement

Part of the original project plan was to subcontract work on the user interface of one particular supplicant, the K Desktop Environment (Linux) “Plasma NM” interface. Since the work was confined to user interface improvements, a contractor with intimate knowledge of KDE and Plasma NM was chosen to execute the work.

In the early stages of execution of the contract however, it turned out that it was impossible to consider the user interface challenges of the KDE supplicant UI in this isolated manner: in order to display e.g. useful error messages for connection failures to the user, the user interface needs to learn about the exact nature of the error from the underlying supplicant and authentication infrastructure.

This underlying infrastructure consists of three separate layers:

- *wpa_supplicant* is the actual supplicant code which authenticates the user to the network
- *NetworkManager* is a connectivity manager which chooses available network connections as required
- *Plasma NM* displays connection states from NetworkManager to users and allows them to configure and make connectivity choices

For the user interface to be able to display meaningful state messages regarding the authentication,

1. *wpa_supplicant* needs to export all the important stages of authentication (the states of the “EAP state machine”) to NetworkManager;
2. *NetworkManager* needs to process and possibly abstract these and
3. pass them on to *Plasma NM*.

After initial investigations, SENSE personnel found out that there are deficiencies on all the layers. *wpa_supplicant* exports some of the required session states via the D-BUS Inter-Process-Communication system, but not all; NetworkManager does not care to consume those D-BUS signals and does not pass them on to user interfaces; the user interfaces then of course cannot display any such information to users.

Since the contracted person is used to writing code near the user interface in KDE’s own code framework, it was not useful to try and get him to work on system-near C code of other unrelated projects. In effect, solving the “better Linux UI” problem in KDE was not possible without a significant re-scope of the work which was not agreed in the project plan, would have needed much more skilled resources than were available, and would be dependent on external factors (third parties agreeing to integrate patches). It would also have taken a long while to get such changes into actual released versions of the software in question, making it an unsuitable target for a short-lived project like SENSE.

The project plan states that the supplicant evaluation work in task 2.2 “may” spin off contracted work. The project decided not to carry out that work. No payment has been made for the preliminary work to the contractor.

4.2 Network configuration beyond EAP

The project has created the draft specification of the EAP metadata configuration format as planned and the specification fulfils its expected role. However, for full supplicant configuration for Wi-Fi purposes, further configuration details need to be transmitted to the device. As one example, the SSID in which the EAP settings are valid need to be communicated.

Commercial examples of proprietary configuration formats simply pack all of the required information into one file (e.g. Apple’s *mobileconfig* format contains Wi-Fi specific settings along with EAP settings, along with IP layer configuration (proxies, static address configuration) and very many other facets of device management).

When seeking standardisation, one needs to consider the borders of responsibility of the standardisation organisations in question. In the case at hand, there is a demarcation line between

the IETF (responsible for EAP, and by corollary also for the manageability of EAP) and the IEEE (responsible for the Wi-Fi standards); possibly also involving industry alliances such as Wi-Fi Alliance.

SENSE was not chartered to and by consequence did not explore standardisation of general Wi-Fi configuration details, nor IP configuration details. As a consequence, the developed Android app is currently hard-wired to assume the SSID "SENSE", encryption with WPA2/AES, automatic IP configuration with DHCP, and no proxy settings. For eduroam purposes, a second version of the app has been created which hard-wires the SSID "eduroam".

In future work, it would be beneficial to define or choose to reuse existing Wi-Fi configuration file formats, and make the developed app consume these extra configuration details so that no separate apps are needed just to be able to cater for different SSIDs.

5 Mapping of results to the requirements of the call text and to the Open Call scope

5.1 Overall Open Call Scope

The scope of the Open Calls is described as:

"The first Open Call for the GN3plus project, now launched, will be devoted to the following goals:

Showcase innovation in the area of data communications and telecommunications by enabling the use of innovative GÉANT network technologies, services and infrastructure.

Achieve maximum market visibility for the technologies and services developed by GÉANT.

Undertake focused work packages that further the continued enhancement and ongoing operation of the leading-edge GÉANT network.

Enhance the combined GEANT [sic] and NRENs ability to provide world class connectivity and services to the knowledge community and to push the state of the art in innovation in Research and Education networking.

It is expected that participants in this Call will propose concrete and specific plans that enable GÉANT to achieve these goals during the given timeframe."

The results of SENSE are a very good match with this general scope of the Open Calls.

SENSE is closely related to eduroam, which (in Europe) is an established GÉANT network service. SENSE improves eduroam:

- By delivering new proposed internet standards (EAP Metadata Configuration file format), it ameliorates the network technologies which underlie the eduroam service.
- By providing verification means of supplicant implementations and RADIUS server behaviour (task 2.2), it directly influences the eduroam infrastructure particularly at its most sensitive spot, the end user interface.
- By having created EAPlab, allows to test-drive eduroam installations (both hardware and software aspects) in a sandbox environment, making the service more easily manageable.

With SENSE's scope of being useful for, but not limited to, eduroam, SENSE has with its manifold public presentations achieved a visibility of its results on the general market of enterprise Wi-Fi authentication. The project has successfully broken out of the containment of the eduroam consortium.

The results of SENSE have a potential for continuation inside the GÉANT network; details can be found in the following chapter "Outlook".

Summarising, the authors believe that SENSE has fully addressed the scope of the Open Calls; that it enhanced the community's ability to provide world class connectivity and services; and that it has pushed the state of the art of innovation in Research and Education networking.

5.2 Specific Call Text of Open Call #16

The call text for Open Call #16 states:

Objectives

The primary objective of the topic is to ensure that all important EAP methods are properly supported by the mostly used supplicants. This topic has the following objectives:

- *To establish a forum of experts setting out minimal requirements for a user-friendly implementation of IEEE 802.1X supplicants, EAP peers and EAP servers. For example, a requirement for a supplicant is that it needs to expose all the parameters to establish unambiguous/mutual authentication to the user; e.g. it is not enough to make the EAP Server's Certification Authority configurable – also the EAP Server's Expected Server Name needs to be configurable. The current user interface for the supplicant in KDE¹ fails this.*
- *To work with as many implementers of IEEE 802.1X / EAP as possible with the goal of improving the existing implementations to meet the minimum requirements. For example, Firefox OS has no User Interface for IEEE 802.1X configuration at all right now, whilst it would be desirable to have one.*
- *To create a knowledge base with tips and checklists for future implementers to lead them to interoperable, user-friendly and complete implementations.*

Expected Impact

The results are expected to raise the usability of IEEE 802.1X and EAP industry-wide to a new level. The project will create a knowledge base with background on, reasoning for, and tips regarding the different aspects of EAP and IEEE 802.1X implementations.

This work also has implications for the usage of the Configuration Assistant Tool (CAT). CAT can only work if the underlying supplicants work well with different EAP types and this is not the case to date. This limits the usage of CAT to a limited number of supplicants and to a limited number of EAP types.

Outputs

- *A reference checklist to describe the minimum level of usability and implementation completeness. This would lead to the creation of a quality label “User-Friendly & Complete IEEE 802.1X supplicant”, “Interoperable EAP server for EAP Types x,y,z”.*
A reference implementation, supported by one or more proof-of-concept(s).
- Improved supplicant software from many implementers of such software.
- The output of this call will be used by GN3plus Joint Research Activity 3 Identity & Trust Technologies for GÉANT Services, specifically by Task 1 (Attributes and Groups).

The SENSE project was created specifically as a response to this open call; naturally, its objectives, impact and produced outputs are a very good match with the call text.

In particular, chapter 3 “Impact” makes a case for the impacts generated (and to be generated in the future) by SENSE. All of the outputs requested by the call except for the quality labels have been tackled by SENSE (a reasoning for the omission of a public quality label for supplicants is given in the task description of task 2.2 in this deliverable; a quality label for EAP servers was not in SENSE’s focus).

5.3 Outlook

There is possible future work in all work areas:

End-User Outreach

The Android app should be improved with new features (e.g. make SSID configurable); apps for other platforms can be envisaged; existing apps with a slightly different scope could be unified or linked more closely (e.g. eduroam Companion).

EAPlab

The developed product should continue to be operated after the end of SENSE and integrated into the eduroam Operations Support Services suite more tightly. Also, the feature set of EAPlab needs continuous development as the industry introduces new requirements (e.g. new hash function or minimum key length warnings in supplicants).

Standardisation

After the end of SENSE, standardisation efforts should continue. One particular aspect was identified which is beyond the IETF scope: Wi-Fi configuration items (SSID, Encryption, ...) are not under IETF change control and their standardisation would need to be sought at other avenues (IEEE, Wi-Fi Alliance, ...).

Supplicant assessments

A future goal is to establish a permanent “EAP Observatory”, where an assessment team evaluates popular eduroam end user devices and informs eduroam NROs, Identity Providers and also end users about the strengths and weaknesses of the various platforms; it also submits bug reports and feature requests to the vendors.

6 Conclusions

SENSE set out “to improve usability of EAP supplicants for end-users, so that they can use Enterprise Wireless Authentication system such as eduroam® easier and with a higher level of security.” The project worked towards this overall objective by improving four distinct areas in the enterprise wireless authentication space:

- Define metrics to assess quality and usability of EAP supplicants; assess supplicants against this metrics [WP 2.2]
- Develop a public test facility to allow testing EAP supplicant behaviour [WP 2.3]
- Standardise configuration of EAP authentication [WP 2.1]
- Improve existing supplicants, including ability to consume said standardised configuration format [WP2.4]

SENSE has fully achieved its goals in all of these areas:

Definition of metrics and supplicant assessment

The project has defined 32 distinct criteria in three categories: Security, Usability and Feature-Completeness. Given their importance, criteria in the Security category were given one of the three weights CRITICAL, MAJOR and MINOR – failure to achieve a CRITICAL/MAJOR requirements led to an immediate “Red/Yellow Card” score. MINOR criteria were treated like the criteria in the other two categories – they were assigned an integer score weight between 1 (less important) and 6 (very important).

Following the criteria and score definitions, the project defined a checklist-style test protocol for the actual assessment of supplicants and assessed a total of seven EAP supplicant implementations against the criteria.

Development of public EAP test facility

SENSE has developed EAPlab, available at <https://eaplab.supPLICANTS.net> EAPlab is an industry-first result: it provides an extremely easy-to-use RADIUS/EAP infrastructure for everybody who is concerned with enterprise wireless (e.g. vendors testing their equipment, administrators studying the behaviour of popular user equipment, bug reporters writing easily reproducible bug reports).

EAPlab’s promise is: point a Wi-Fi Access Point to EAPlab's RADIUS server infrastructure and an entire roaming consortium will be at your disposal.

EAPlab allows the simulation of working, misconfigured and malign EAP Identity Providers (IdPs) with several RADIUS server implementations, and with multiple EAP types (incl. client certificate based). Switching between all the variations is done with a single click. Every logged in user of EAPlab can thus quickly verify how an EAP supplicant will behave in these non-standard situations.

Standardisation

SENSE created two revisions of an IETF Internet-Draft for an EAP Configuration file format. The drafts (individual submission, document track: standards) were submitted prior to and discussed in the meetings IETF89 (U.K.) and IETF91 (U.S.A.). Configuration files according to this draft specification are produced by eduroam CAT (version 1.1) and are consumed by the eduroam CAT Android app and PSNC's Linux eduroam installer.

Supplicant improvements

Subcontractor Swansea University has developed the “eduroam CAT” Android app – it consumes EAP configuration files, and uses Android API to provision all EAP configuration details into the device. The app requires API level 18 (Android 4.3 and above), and is publicly available on the Google Play Store since 27 Mar 2015.

PSNC have written a Linux installer that also consumes EAP configuration files and uses D-Bus signalling to provision the configuration details into numerous Linux supplicant user interfaces and back ends.

SENSE has delivered significant value for money, and there are many aspects in the project which lead to possible new and continuing work, as described in the Outlook chapter. A follow-up project to SENSE would make – sense.

Appendix A Android eduroam CAT: IdP Admin Guide

A.1 Introduction

Configuring eduroam securely on Android devices has proven difficult for users, with many users setting the minimum settings in order to connect, but not fully configuring eduroam to enable complete security. The eduroam CAT app aims to solve this problem by securely configuring eduroam connections for users, while keeping usability easy.

Android will not run untrusted apps by default, so using the Google Play Store to distribute the app to users is required. This means having one generic configuration app that can be customized to configure each institutions users. eap-config files are used to distribute the custom configuration information, and the eduroam CAT app natively understands these.

The eap-config files can either be hosted locally by an institution, or users can be directed to the cat.eduroam.org website to discover their eap-config files.

The app requires a minimum of Android 4.3 (API 18).

A.2 The eduroam CAT App

The eduroam CAT app has three sections:

1. **Install** – Used to install a profile to the device, using credentials set by the user. This section will also provide feedback on the state of the currently installed eduroam profile. See figure 1.
2. **Profiles** – Used to display information about the currently installed profile. This section will also direct users to your home institution page or cat.eduroam.org for profile discovery. See figure 2.
3. **Status** – Generic wireless and supplicant status information. This section also provides the option to dig deeper into any problems connecting, using the debug option. See figure 3.

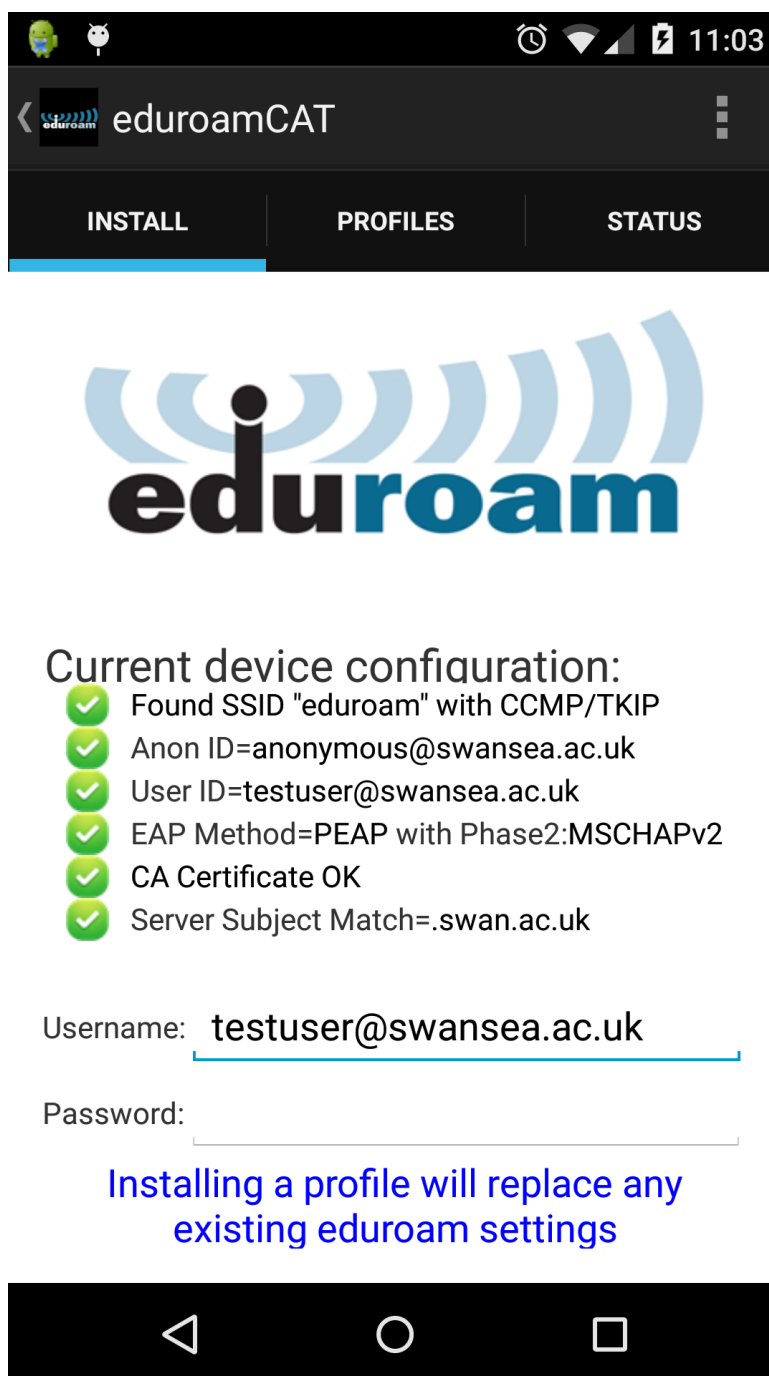


Figure 1 – Install Section

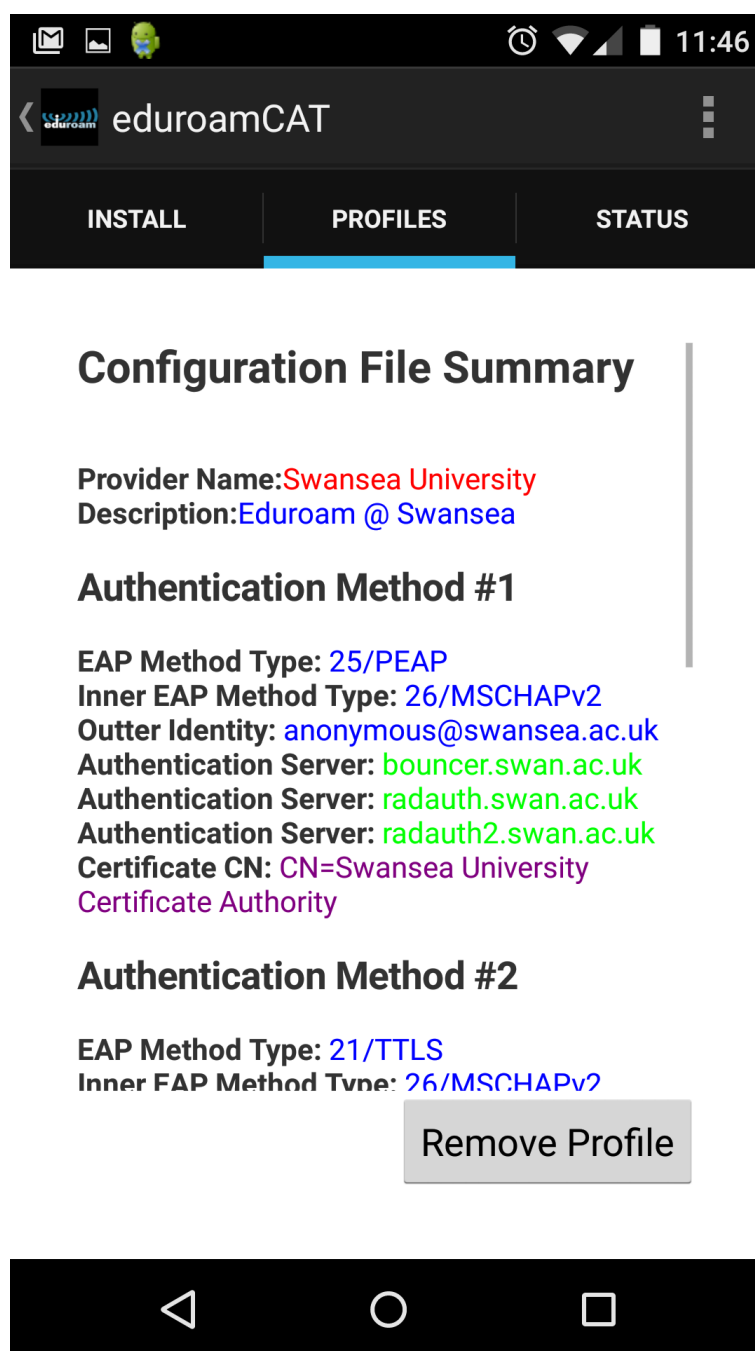


Figure 2 – Profiles Section

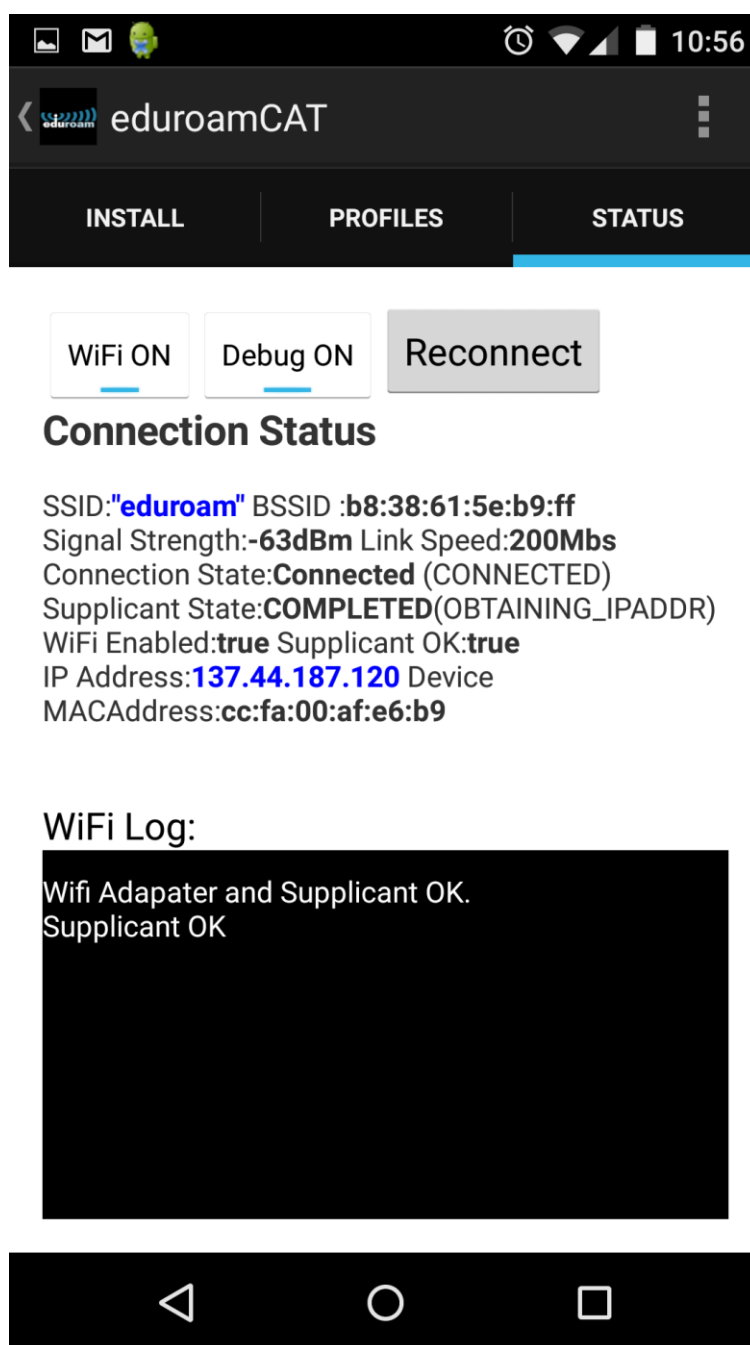


Figure 3 – Status section

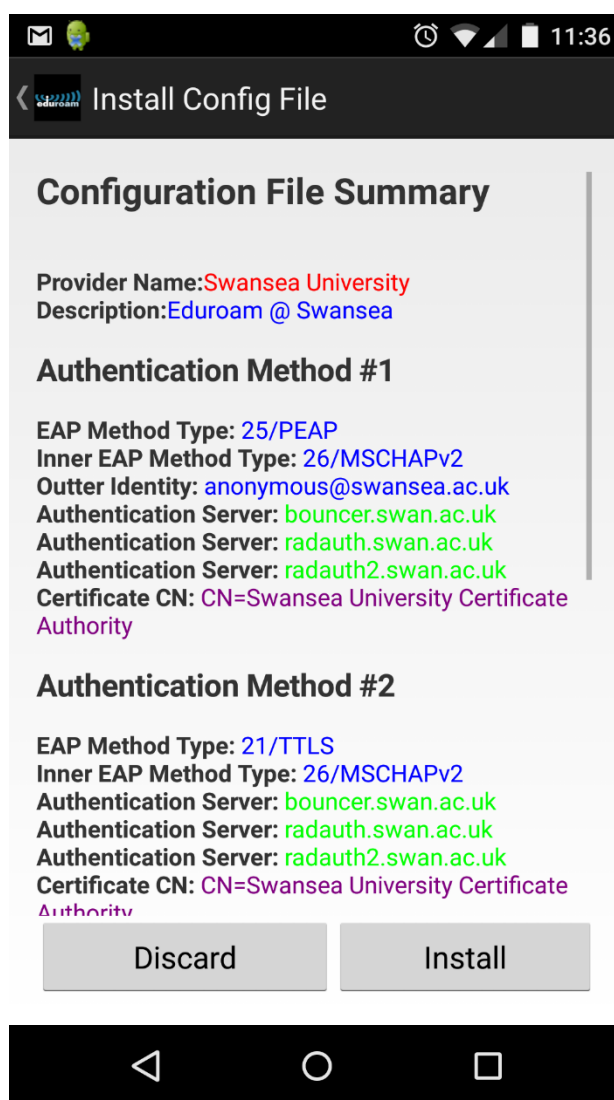


Figure 4 – Install Profile Section

Once a user clicks on an eap-config file, the app will then load an install page, showing the provider information, support information and supported Authentication Methods.

Authentication methods are used in ascending numerical order.

Installing a profile at this stage does not connect the user to eduroam, but install the profile into the app, so the user can then enter their credentials to connect.

A.3 The eap-config File

The eap-config files have the extension of “.eap-config”

If you host the file locally, you need to set the web server to understand the MIME type:

# MIME type	Extensions
application/eap-config	eap-config

In Linux/Apache installs this is typically in: **/etc/mime.types**

The eap-config files contain all the information needed by the app to configure a institutions user for eduroam, and provide support information.

The cat.eduroam.org website will provide users of CAT with automatically generated eap-config files.

A.4 Play Store and Captive Portals

To allow access to the app through a captive portal network, you need to open the following HTTPS web sites up through a captive portal:

1. <https://play.google.com>
2. <https://gstatic.com>
3. <https://apis.google.com>
4. <https://ggpht.com>

A.5 Example: Swansea University

An example eap-config can be found here: <http://swis.swan.ac.uk/swansea.eap-config>

```
<?xml version="1.0" encoding="utf-8"?>
<EAPIdentityProviderList
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="eap-metadata.xsd">
<EAPIdentityProvider ID="XXX" namespace="urn:RFC4282:realm" lang="en"
version="1">
<AuthenticationMethods>
<AuthenticationMethod>
<EAPMethod>
<Type>25</Type>
</EAPMethod>
<ServerSideCredential>
```

```
<CA format="X.509" encoding="base64">
MIIFFDCCA/ygAwIBAgIJAKHEgEnPtOG5MA0GCSqGSIb3DQEBBQUAMIG2MQswCQYDVQQG
EwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5zZWExGzAZBgNVBAoTElN3
YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJKoZIhvcNAQkBFhwhb3N0
bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5zZWEGVW5pdmVyc2l0eSBD
ZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwggEiMA0GCSqGSIb3DQEB
AQUAA4IBDwAwggEKAoIBAQC8355bEld4lTJWllix5Q/RCymxS3gsPMRYS/dCyEeXpXG6
oQvkl1s1bFwMHJq9JGo/5/JPVXS1Pcn8yOhZ3OYHmQyzRyhPHiqn0tMhwpLDmCMUeV2tj
BLlb4ipOzaLrCfczDEbXB+5YFu/K7AsxYoGwFNNRzC3wv5H6+py4LuRwRdvBzVHFWaEi
KzdYuyNcDMzXoRs4vftvd91EzZCDejyFg3tGW2xHCUJmg/4gSO+Wl63Hhc2bbzS3/ZJZ
7SJlFas5TL5FEiMR9vkslEbDVGGG1sFRJb6dUYfDxOAbMgI+flqXETp6rnh2C2WCgwhk
uaDbd/JeTyITfdMEMcPu6OTfAgMBAAGjggEfMIIBGzAdBgNVHQ4EFgQUaE9czZ3lJ6WU
dHLQP1K5/2bKI/0wgesGA1UdIwSB4zCB4IAUAe9czZ3lJ6WUdHLQP1K5/2bKI/2hgbyk
gbkwgbYxCzAJBgNVBAYTAkdCMQ4wDAYDVQQIEwVXYWxlczEQMA4GA1UEBxMHU3dhbnNl
YTEbMBkGA1UEChMSU3dhbnNlYSBVbml2ZXJzaXR5MjE1MjE1MjE1MjE1MjE1MjE1MjE1
YSBVbml2ZXJzaXR5MjE1MjE1MjE1MjE1MjE1MjE1MjE1MjE1MjE1MjE1MjE1MjE1MjE1
EwQFMAMBAf8wDQYJKoZIhvcNAQEFBQADggEBABT9P2Yz0hpJwA69AjLVjznI3pbgRUZU
7AnNgQyzB21Jr0LVbTcMADEF53eOVHTV2fn5GEre83oCmVH14xnu7LTUk/sIuLVznwbq
nY+k6JHjz5NSKhywhdilOb5FPoSStPbPOiWKDkl5JI6S4RYEXCtuVhAFQCykdyea0JpJ
5tdqa59rvbLrTQx/pVGnQrIWcoS6UXbmAly0tiff6k6zPeX/3c+Kglo0QakuRhQt/8Ak
12eugirgYtuakk3x8tWvnWSJuJluPFyw7V+X+mXeAwjro2AzojAr93qFSzZcO3q5hYNq
/rjHGq9lVMfNsbjdiE2IM1OnM8cj5nJpmPD6uJQ=
</CA>
<ServerID>bouncer.swan.ac.uk</ServerID>
<ServerID>radauth.swan.ac.uk</ServerID>
<ServerID>radauth2.swan.ac.uk</ServerID>
</ServerSideCredential>
<ClientSideCredential>
<OuterIdentity>anonymous@swansea.ac.uk</OuterIdentity>
</ClientSideCredential>
<InnerAuthenticationMethod>
<EAPMethod>
<Type>26</Type>
</EAPMethod>
</InnerAuthenticationMethod>
</AuthenticationMethod>
<AuthenticationMethod>
<EAPMethod>
<Type>21</Type>
</EAPMethod>
<ServerSideCredential>
<CA format="X.509" encoding="base64">
MIIFFDCCA/ygAwIBAgIJAKHEgEnPtOG5MA0GCSqGSIb3DQEBBQUAMIG2MQswCQYDVQQG
EwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5zZWExGzAZBgNVBAoTElN3
YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJKoZIhvcNAQkBFhwhb3N0
bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5zZWEGVW5pdmVyc2l0eSBD
ZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MITYxNDMw
MTFamIG2MQswCQYDVQQGEwJHQjEOMAwGA1UECBMFV2FsZXMxEDAOBgNVBACTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMMAoGA1UECXMDSVNTMScwJQYJ
KoZIhvcNAQkBFhwhb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0
```



```
MTFaMIG2MQswCQYDVQQGEwJHQQjEOMAwGA1UECBMFV2FsZXMxEDAObGNVBAcTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMAoGA1UECzMDSVNTMScwJQYJ
KoZIHvcNAQkBFhwb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwgGElMA0GCSqGSIb3DQEB
AQUAA4IBDwAwggEKAoIBAQC8355bEld4lTJWlllX5Q/RCymxS3gsPMRYS/dCyEeXpXG6
oQvkl1sJbFwMHJq9JGo/5/JPVXS1Pcn8yOhZ3OYHmQyzRyhPHiqn0tMhwpLDmCMUeV2tj
BLlB4ipOzaLrCfczDEbXB+5YFu/K7AsxYoGwFNNRzC3wv5H6+py4LuRwRdvBzVHFWaEi
KzdYuyNcDMzXoRs4vftvd91EzZCDejyFg3tGW2xHCUJmg/4gSO+Wl63Hhc2bbzS3/ZJZ
7SJlFas5TL5FEiMR9vkslEbDVGGG1sfRJB6dUYfDxOAbMgI+f1qXetp6rnh2C2WCgwhk
uaDbd/JeTyITfdMEmcPu6OTfAgMBAAGjggEfMIIBGzAdBgNVHQ4EFgQUaE9czZ3lJ6WU
dHLQP1K5/2bKI/0wgesGA1UdIwSB4zCB4IAUaE9czZ3lJ6WUdHLQP1K5/2bKI/2hgbyk
gbkwgbyXcZAJBgNVBAYTAkdCMQ4wDAYDVQQIEwVXYWxlczEQMA4GA1UEBxMHU3dhbnNl
YTEbMBkGA1UEChMSU3dhbnNlYSBvbm12ZXJzaXR5MQwwCgYDVQQLEwNlU1MxJzAlBgkq
hkiG9w0BCQEWGHBvc3RtYXN0ZXJAc3dhbnNlYS5hYy51azExMC8GA1UEAxMoU3dhbnNl
YSBvbm12ZXJzaXR5IENlcnRpZml1eXJlIEF1dGhvcml0eYIJAkHEGEnPtOG5MAwGA1Ud
EwQFMAMBAf8wDQYJKoZIhvcNAQEFBQADggEBABT9P2Yz0hpJwA69AjLVjznI3pbGRUZU
7AnNgQyzB21Jr0LVbTCmADEF53eOVHTV2fn5GEre83oCmVH14xnu7LTUk/sIuLVznwbq
nY+k6JHjz5NSKhywhdilOb5FPoSStPbPOiWKDkl5JI6S4RYEXCtuVhAFQCykdyea0JpJ
5tdqa59rvbLrTQx/pVGnQrIWcoS6UXbmAlY0tiff6k6zPeX/3c+Kglo0QakuRhQt/8Ak
12eugirgYtuakk3x8tWvnWSJuJ1uPFyw7V+X+mXeAWjro2AzojAr93qFSzZcO3q5hYNq
/rjHGq9lVMfNsbjdiE2IM1OnM8cj5nJpmPD6uJQ=
```

</CA>

<ServerID>bouncer.swan.ac.uk</ServerID>

<ServerID>radauth.swan.ac.uk</ServerID>

<ServerID>radauth2.swan.ac.uk</ServerID>

</ServerSideCredential>

<ClientSideCredential>

<OuterIdentity>anonymous@swansea.ac.uk</OuterIdentity>

</ClientSideCredential>

<InnerAuthenticationMethod>

<EAPMethod>

<Type>26</Type>

</EAPMethod>

</InnerAuthenticationMethod>

</AuthenticationMethod>

<AuthenticationMethod>

<EAPMethod><Type>21</Type></EAPMethod>

<ServerSideCredential><CA format="X.509" encoding="base64">

```
MIIFDCCA/ygAwIBAgIJAKHEGEnPtOG5MA0GCSqGSIb3DQEBBQUAMIG2MQswCQYDVQQG
EwJHQQjEOMAwGA1UECBMFV2FsZXMxEDAObGNVBAcTB1N3YW5zZWExGzAZBgNVBAoTElN3
YW5zZWEGVW5pdmVyc2l0eTEMMAoGA1UECzMDSVNTMScwJQYJKoZIHvcNAQkBFhwb3N0
bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5zZWEGVW5pdmVyc2l0eSBD
ZXJ0aWZpY2F0ZSBBdXRob3JpdHkwIBcNMTAwOTA5MTQzMDExWhgPMjExMDA4MTYxNDMw
MTFaMIG2MQswCQYDVQQGEwJHQQjEOMAwGA1UECBMFV2FsZXMxEDAObGNVBAcTB1N3YW5z
ZWExGzAZBgNVBAoTElN3YW5zZWEGVW5pdmVyc2l0eTEMMAoGA1UECzMDSVNTMScwJQYJ
KoZIHvcNAQkBFhwb3N0bWFzdGVyQHN3YW5zZWEuYWMudWsxMTAvBgNVBAMTKFN3YW5z
ZWEGVW5pdmVyc2l0eSBDZXJ0aWZpY2F0ZSBBdXRob3JpdHkwgGElMA0GCSqGSIb3DQEB
AQUAA4IBDwAwggEKAoIBAQC8355bEld4lTJWlllX5Q/RCymxS3gsPMRYS/dCyEeXpXG6
oQvkl1sJbFwMHJq9JGo/5/JPVXS1Pcn8yOhZ3OYHmQyzRyhPHiqn0tMhwpLDmCMUeV2tj
BLlB4ipOzaLrCfczDEbXB+5YFu/K7AsxYoGwFNNRzC3wv5H6+py4LuRwRdvBzVHFWaEi
KzdYuyNcDMzXoRs4vftvd91EzZCDejyFg3tGW2xHCUJmg/4gSO+Wl63Hhc2bbzS3/ZJZ
7SJlFas5TL5FEiMR9vkslEbDVGGG1sfRJB6dUYfDxOAbMgI+f1qXetp6rnh2C2WCgwhk
```

```

uaDbd/JeTyITfdMEmcPu6OTfAgMBAAGjggEfMIIBGzAdBgNVHQ4EFgQUaE9czZ3lJ6WU
dHLQP1K5/2bKI/0wgesGA1UdIwSB4zCB4IAUaE9czZ3lJ6WUdHLQP1K5/2bKI/2hgbyk
gbkwgbyYxCzAJBgNVBAYTAkdCMQ4wDAYDVQQIEwVXYWxlczEQMA4GA1UEBxMHU3dhbnNl
YTEbMBkGA1UEChMSU3dhbnNlYSBVbml2ZXJzaXR5MjEwYzU1aW50YzU1aW50YzU1aW50
hkiG9w0BCQEWGHBvc3RtYXN0ZXJAc3dhbnNlYS5hYy51azExMC8GA1UEAxMoU3dhbnNl
YSBVbml2ZXJzaXR5IENlcnRpZmljYXRlIEF1dGhvcml0eYIJAKHEgEnPtOG5MAwGA1Ud
EwQFMAMBAf8wDQYJKoZIhvcNAQEFBQADggEBABT9P2Yz0hpJwA69AjLVjznI3pbgrUZU
7AnNgQyzB21Jr0LVbTCmADEF53eOVHTV2fn5GEre83oCmVH14xnu7LTUk/sIuLVznwbq
nY+k6JHjz5NSKhywhdilOb5FPoSStPbPOiWKDkl5JI6S4RYEXCtuVhAFQCykdyea0JpJ
5tdqa59rvbLrTQx/pVGnQrIWcoS6UXbmAlY0tiff6k6zPeX/3c+Kglo0QakuRhQt/8Ak
12eugirgYtuakk3x8tWvnWSJuJ1uPFyw7V+X+mXeAWjro2AzojAr93qFSzZcO3q5hYNq
/rjHGq9lVmfNsbjdIE2IM1OnM8cj5nJpmPD6uJQ=
</CA>
<ServerID>bouncer.swan.ac.uk</ServerID>
<ServerID>radauth.swan.ac.uk</ServerID>
<ServerID>radauth2.swan.ac.uk</ServerID>
</ServerSideCredential>
<ClientSideCredential>
<OuterIdentity>anonymous@swansea.ac.uk</OuterIdentity>
</ClientSideCredential>
<InnerAuthenticationMethod>
<NonEAPAuthMethod>
<Type>1</Type>
</NonEAPAuthMethod>
</InnerAuthenticationMethod>
</AuthenticationMethod>
</AuthenticationMethods>
<ProviderInfo>
<DisplayName>Swansea University</DisplayName>
<Description>Eduroam @ Swansea3</Description>
<ProviderLocation>
<Longitude>-3.978525300000001</Longitude>
<Latitude>51.610126</Latitude>
</ProviderLocation>
<ProviderLocation>
<Longitude>-3.97838139999999893</Longitude>
<Latitude>51.6102499</Latitude>
</ProviderLocation>
<TermsOfUse>No terms of use yet...</TermsOfUse>
<Helpdesk>
<EmailAddress>wirelessSupport@swansea.ac.uk</EmailAddress>
<WebAddress>http://swis.swan.ac.uk</WebAddress>
<Phone>00441792602235</Phone>
</Helpdesk>
</ProviderInfo>
</EAPIIdentityProvider>
</EAPIIdentityProviderList

```