

学認連携 Moodle での受講情報確認方法の実装について

浜元 信州^{1,a)} 横山 重俊^{1,3} 上田 浩^{2,3}

概要: 国立情報学研究所で「りんりん姫」等の教材を提供している学認連携 Moodle では、個人 ID が匿名化されているため、利用機関での受講情報の確認には、匿名化された情報の再識別が必要となる。我々は、共同利用 LMS での匿名化された ID を再識別し、各機関でデータを利用する方法を検討し、りんりん姫での成績確認や受講履歴の確認を例として、各大学で利用可能な成績確認システムを開発した。本発表では、本システムについて紹介するとともに、今後の改良について展望を述べる。

キーワード: 学認連携 Moodle, Shibboleth

On the confirmation method of the user's record of the Gaku-Nin Moodle

HAMAMOTO, NOBUKUNI^{1,a)} YOKOYAMA, SHIGETOSHI^{1,3} UEDA, HIROSHI^{2,3}

Abstract: The Gaku-Nin moodle provides the teaching materials for information ethics called "The princess Rin-RIn" to the universities through the academic access federation in Japan named "Gaku-Nin". In order to confirm user's record on the Gaku-Nin moodle, the administrator of university needs re-identify anonymized user id saved on the Gaku-Nin moodle. We discuss and compare the methods to reidentify anonymized user id and we report the cyberethics grade confirmation system which is used by the university administrator to confirm user's record. In the present paper, we introduce the cyberethics grade confirmation system and discuss planned improvements of the system.

Keywords: Gaku-Nin Moodle, Shibboleth

1. はじめに

情報セキュリティ教育を徹底することは、近年、多くの高等教育機関にとっての課題である。国立情報学研究所では、この課題に応えるため、「高等教育機関の情報セキュリティ対策のためのサンプル規程集 (以下サンプル規程集)」[1] を策定し、サンプル規程集に準拠した情報倫理教育のための e ラーニング教材を、学認連携 Moodle 講習サイト

(<https://security-learning.nii.ac.jp>) において全国の学認参加機関に無償提供している [2], [3]。

学認連携 Moodle には、各大学の活動の履歴、成績など機微な情報が集約されるため、情報漏えいのリスクも高くなる。各大学での共同利用 LMS へ参加同意を得ることが難しくなるほか、共同利用 LMS 自身にも厳重なセキュリティ対策を強いることになるため、学認連携 Moodle 運用上の課題の一つとなっている。学認では、この問題を解決するため、匿名 ID (eduParsonTargetedId: ePTID) で ID を匿名化することで情報漏えいの影響を軽減している。大学等参加機関の IdP (ID Provider: 大学内認証サーバ) は、共同利用 LMS に対して、氏名、所属などの個人情報を送信せず、匿名 ID のみを送信する。学認連携 Moodle でも、この方式を採用し、100 を超える高等教育機関での大規模な共同利用を実現した [2]。しかし、匿名 ID では、

¹ 群馬大学総合情報メディアセンター
Library and Information Technology Center, Gunma University, 4-2 Aramaki-machi, Gunma 371-8510, Japan

² 京都大学学術情報メディアセンター
Academic Center for Computing and Media Studies, Kyoto University

³ 国立情報学研究所
National Institute of Informatics

a) n.hamamoto@gunma-u.ac.jp

各大学の教師が学生個人を特定できないため、担当する授業の受講生の成績を見る等の管理ができない。

この問題を解決するため、我々は、大学の IdP と連携して匿名 ID から個人を特定し、教師に学生の受講情報を提供する「受講情報確認システム」を提案し、開発を行ってきた [4], [5], [6]。開発されたソフトウェアは、群馬大学で利用され、現在公開されている [7]。現状では、成績の入力は、学認連携 Moodle からダウンロードした CSV ファイルを、本システムに登録することとしているが、本運用では、管理者負担が大きいという問題が残る。本論文では、成績連携の手法の改良に向けた検討結果を示し、実装方法の概要を記載する。

2. 受講情報確認システム

受講情報確認システムは、学認連携 Moodle 上でサービスされている「りりん姫」の総合テストの成績、受講履歴を各大学で確認するためのシステムである。この章では、本システムの開発の背景とシステム概要を述べる。

2.1 システム開発の背景

本システム開発の背景として、群馬大学での「りりん姫」の利用方法を述べ、本システムの必要性と要件を記載する。

群馬大学での利用方法としては主に 2 つあり、一つは、学生を対象とした情報セキュリティ教育である。具体的には、初年度教育の一環として「りりん姫」を利用しており、群馬大学の学部生全てが受講する必修の授業「情報」で、本最終テストへの合格を単位取得の必須要件としている。

もう一つは、主に教職員を対象とした情報セキュリティ教育である。具体的には、「りりん姫」の受講を VPN 接続と eduroam の利用条件とし、総合テストの合格者のみが、両サービスを利用できるようにしている。このような形で、全学学生、教職員の情報セキュリティレベルの底上げを行っているが、この運用を行うために必須のシステムが、受講情報確認システムである。

上記の運用を行うためには、学内者が成績の確認を行うことが必須要件となる。「りりん姫」は、HTML5 でのコンテンツと、最終テストで構成された情報セキュリティ教材であり、最終テストには、コンテンツを全て受講しないと最終テストが受けられないよう構成されている。このため、最終テストの点数を学内で確認できなくてはいけない。具体的な最終テストの確認を行う者は、教職員の最終テストの場合は、総合情報メディアセンターの担当者、学生の場合は、授業担当教員である。また、各クラスの担当教員はクラスの学生の成績のみを見ることができ、他は見られないよう構成する必要がある。

2.2 システム概要

上記の「りりん姫」の利用は群馬大学を例としているが、他の大学でも、りりん姫を利用した授業展開を行う場合には同様の要件が発生する。この要件を考慮して大学内の状況をモデル化した受講情報確認システム [4] の概要を図 1 に示す。本システムは、以下の権限を持ったユーザーで構成される。

(1) IdP 管理者

大学の IdP を管理し、ePTID と大学の user ID (uID) の対応に関する情報を管理している。

(2) 教務システム管理者 (クラス管理者)

クラスの構成員と担当教員の情報を管理している。

(3) 教員 (クラス成績確認担当者)

担当クラスの構成員の成績を確認する。

(4) 受講情報確認システム管理者

受講情報確認システムの管理を行う。学認連携 Moodle 大学管理者と同じとする。

(5) 全学成績確認担当者

全学の構成員の成績を管理する。

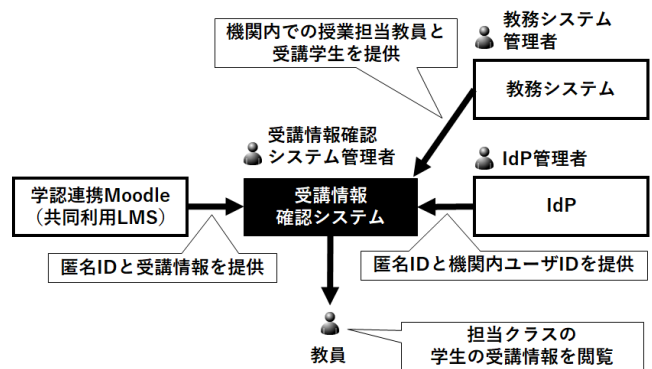


図 1 受講情報確認システムの概要図

群馬大学の例でいうと、教員は各クラスの授業担当教員を想定しており、全学成績確認担当者は、総合情報メディアセンターの担当者を想定している。また、教員の成績閲覧を担当クラスに限定する必要があるため、クラスの構成員と担当教員の情報を登録する教務システム管理者が必須となる。

2.3 成績確認方式

本システムを用いて、成績の確認を行うが、成績の確認を行うためには、「りりん姫」の受講情報が蓄積されている学認連携 Moodle から成績情報を取得する必要がある。ここでは、成績確認方式として 3 種類の方式を提案し、それぞれの特徴を述べ、比較を行う。

2.3.1 確認方式 (1)

現時点で、学認連携 Moodle から成績を取得するには、受講情報確認システム管理者が、学認連携 Moodle から、成

績情報を CSV ファイルでダウンロードするしかない。このため、ダウンロードした結果を、受講情報確認システムに登録するというのが、最も単純に考えられる方法である。受講情報確認システムの現在のバージョンではこの方法を実装している。学認連携 Moodle では、受講者は ePTID で記載されているため、そのままでは、学生を特定することができない。このため、受講情報確認システムでは、IdP へ ePTID とユーザ ID の対応を問い合わせている。この方式を採用することで、ePTID とユーザ ID の対応関係を取ることができるが、いくつかの問題点がある。

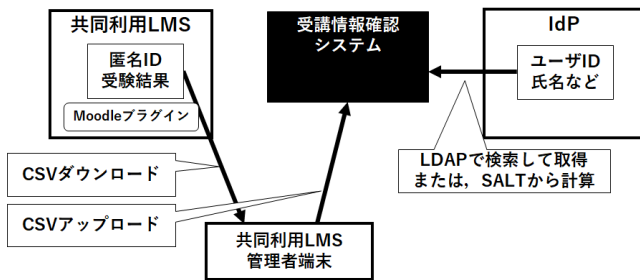


図 2 確認方式 (1) の概要図

Fig. 2 Overview of the confirmation method (1)

確認方式 (1) の問題点

- (1) 受講情報確認システム管理者負担が大きい
 本方式では、受講情報確認システムの管理者が、学認連携 Moodle から、受講情報を CSV 形式で取得して登録する。この方式では、管理者が手動で登録を行うため、管理者の負担が大きい。
- (2) 学認連携 Moodle の受講結果をリアルタイムで反映できない
 上記の問題点と関連するが、CSV 登録を手動で行っている以上、受講生の受講結果が、受講情報確認システムに即時反映されることはない。反映のタイミングは、管理者が作業を行ったタイミングとなる。
- (3) PersistentID にするか、SALT の提供の必要がある
 本方式では IdP から ePTID とユーザ ID の対応を取得する。IdP では ePTID とユーザ ID の対応関係を知っていることが前提となっている。Shibboleth IdP では ePTID の生成を、Computed ID と呼ばれる方法、または、Stored ID と呼ばれる方法により行っている。Computed ID では、IdP 毎に管理者が決めた SALT を利用して、接続する SP の entityID、ユーザ ID、SALT をハッシュ化して ePTID を先生している。Stored ID でも同様であるが、万が一の衝突を防ぐことができると言われており、IdP 上にユーザ ID、接続 SP、ePTID が属性となるテーブルを保存している。IdP での ePTID の実装が、StoredID で構築されている場合には、受講情報確認システムから、ePTID を指定して、ユーザ ID を、SQL で検索することができ

る。これには、IdP 管理者の許可が必要な点が問題点の一つであり、受講情報確認システムの管理者と、IdP 管理者の間で、検索権限に関して合意形成する必要がある。Computed ID の場合には状況はより深刻で、ePTID からユーザ ID を取得する方法はない。(取得できる場合には、ePTID の匿名性がそもそも成り立たない) このため、受講情報確認システム側では、IdP から SALT 自身を取得する必要がある。SALT がわかれば、学認連携 Moodle 以外の SP に対して ePTID を計算することが可能となるため、IdP 管理者との合意形成は非常に難しくなる。SALT 提供に関する問題が解決する例としては、小規模の大学では、IdP も受講情報確認システムも同一の管理者が運用する時がある。幸いにして、多くの大学が管理者が分かれていない状況になっているのではないかと推測しており、この問題のインパクトは小さいかもしれない。しかし、管理局が IdP は、情報基盤を管理する部局、受講情報確認システムは情報教育を管理する部局が管理等、両管理者が別人/別部局というケースも十分ありうるため、対策を練る必要はある。

- (4) 受講情報確認システムに対するユーザ成績送信同意を取得していない

受講情報確認システムを利用して成績を管理を行うためには、当然ではあるが、学認連携 Moodle にある成績情報を受講情報確認システムに送信する必要がある。ユーザ情報を別システムに送信する際には、原則としてユーザ本人の同意を取った方が良いが、確認方式 (1) では特に同意を得ているわけではないという点が問題点の一つである。この問題点は「ユーザの成績を先生が見る」という行為は、特に説明がなくともユーザの承諾している事項であり、あえて了承を取る必要はないとする考え方もある。その場合にはこの点は考慮する必要はない。

2.3.2 確認方式 (2)

上記の確認方式 (1) の問題点を踏まえて、新たな確認方法を考える。特に問題点 (1), (2) に注目して、下記のような確認方法を提案する：

- 受講情報確認システムと学認連携 Moodle との通信を定期的に行い、受講情報を同期する。

同期の方法をここでは指定していないが、下記のような方法が考えられる。

- (1) 学認連携 Moodle 管理者のアカウントを利用し、定期的にログイン、指定の情報を取得するスクリプトを定期的に行う。
- (2) 学認連携 Moodle 上のデータベースにアクセス可能なアカウントを作成し、定期的に SQL を実行する。
- (3) 受講情報確認システムに対して、外部からアクセス可能なアカウントを作成し、学認連携 Moodle 上のデー

データベースに定期的に成績を書き込む。
上記のどの方法をとっても、成績が定期的に自動同期されるため、管理者の手を煩わすことはなくなる。このため問題点 (1) は解決する。問題点 (2) は定期実行となるため、厳密な意味では解決しない。ただし、同期間隔を短くすることにより、業務に支障を出さないような遅延に収めることは可能になると考えられる。しかしながら、どの方法を取った場合でも実装が難点があること分かっており、下記に詳細を示す。

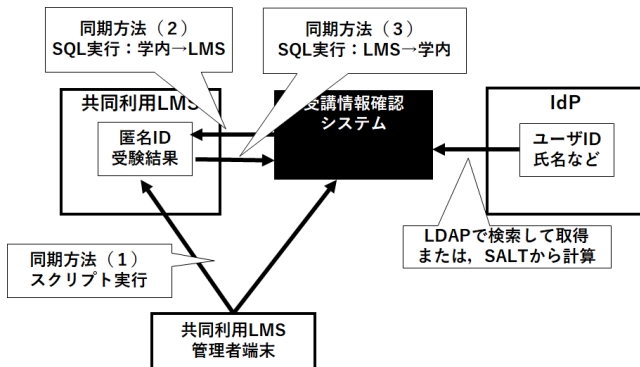


図 3 確認方式 (2) の概要図
Fig. 3 Overview of the confirmation method (2)

確認方式 (2) の問題点

(1) 同期方法 (1) を取った場合

この場合には、スクリプトの作成の際にパスワードを必要とする。このパスワードをどのようにスクリプトに覚えさせるかについては、別途方法を考慮する必要がある。また、スクリプト自体の作成も、Moodle 自体の手動操作をシミュレートする必要が発生し、簡単ではない。

(2) 同期方法 (2) を取った場合

この場合、学認連携 Moodle 上に、機関毎のユーザを作成する必要があるため、学認連携 Moodle 上でのアカウント管理が煩雑になる点が問題である。また、学認連携 Moodle の SQL に対してのアクセス制限に対しては、機関毎、つまり、アカウント毎に制限する等の対策が必要となる。管理を厳密にするには、機関の受講情報確認システムの IP アドレス等の情報を入手する必要があり、管理はますます煩雑である。また、アカウント毎の、検索制限を設ける必要もあり、アカウント毎のビューを作成するなどして実現する必要があるが、この管理も煩雑であり、学認連携 Moodle 管理者に多大な負担をかける方式であるといえる。現状の CSV ファイル取得の際にも、SQL 実行には負荷がかかっており、頻繁に同期する可能性も考えると学認連携 Moodle 側のサーバ資源も問題となる。

(3) 同期方法 (3) を取った場合

この場合には、受講情報確認システムに対して学認連

携 Moodle からアクセス可能なように、ポート開放する必要はある。多くの機関では、ファイアウォールによるアクセス制限を行っており、学外からのアクセス許可を行うための手続きを取る必要がある。

以上のように各手法に対しても固有の問題点があるほか、どの場合でも、学認連携 Moodle に対して負担をかけることになるという共通の問題点がある。現在、学認連携 Moodle でりんりん姫の成績、受講履歴を取得する場合にはかなりの時間がかかる。学認連携 Moodle で扱うユーザ数が膨大になっていることが原因の一つであり、これを解消するため、昨年度から学認連携 Moodle の年次リセットが開始となった。年次リセットにより、この問題は改善されたが、いずれにしても、今後も利用機関が増えることを見込むと、改善すべき問題である。

2.3.3 確認方式 (3)

確認方式 (3) では、各ユーザから受講情報を受講情報確認システムに送信してもらうことを考える。

(1) 各ユーザは、受講情報確認システムに対して、ユーザの成績を連携する。

連携の方法は、各ユーザに受講情報確認システムにログインして成績を入力してもらうことも考えられるが、この方法では、成績の正当性を受講情報確認システムの管理者が再度確認する必要があり、方式として成り立たない。ここでは、Shibboleth 認証 [8] の認証の流れと同様に、ブラウザを介して、成績情報を取得することを考える。すなわち、Shibboleth 認証の際に、IdP から SP に属性を渡すのと同様に、学認連携 Moodle から受講情報確認システムに対して成績を渡す方法を提案する。この方式では、Moodle プラグインを開発し、下記の成績やりとりの流れを実現する。

- (1) ユーザは、機関毎に準備された、学認連携 Moodle コース上の成績報告ボタン等を押す、成績を報告する。
- (2) 成績報告ボタンを押すことにより、学認連携 Moodle 上の DB を検索して、そのユーザの成績と ePTID をブラウザ経由で、受講情報確認システムに送信
- (3) 受講情報確認システムにユーザがログインし、学認連携 Moodle のユーザ ID 属性を得る。
- (4) ブラウザからの送信を受けて、受講情報確認システムへ ePTID と成績を登録

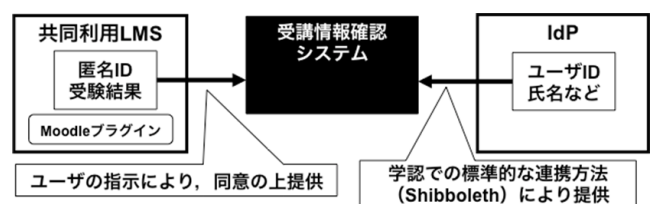


図 4 連携方法 (3) の概要図
Fig. 4 Overview of the confirmation method (3)

この方法により、確認方式 (1) で指摘した問題点の多く

が解消される。この方式では、確認方式(1)にあったような受講情報確認システムの管理者の手動操作は必要とならないため、問題点(1)の受講情報確認システムの管理者負担は大幅に削減される。

確認方式(1)の問題点(3)にあった、IdPとのやり取りは、本方式では不要となる。実際には、受講情報確認システムへのログインを必要とするので、受講情報確認システムとIdPとのやり取りは発生しているのだが、通常の Shibboleth 認証の範囲内の話しである。

確認方式(1)の問題点(4)にあったユーザ同意の取得については、Moodle プラグイン上で、成績を送信する前に、ユーザに同意を取得することで解決することができる。

確認方式(1)の問題点(2)については、解決は難しい。ユーザが成績を報告すれば、即座に受講情報確認システムに反映されるため、その意味ではリアルタイムであるが、ユーザが成績を報告し忘れる等の新たな問題点が発生する可能性がある。この点はプラグイン開発でUIを工夫する必要がある。

確認方式(3)の問題点

● 受講情報システムの学外公開

この方式では、成績の連携はユーザのブラウザを経由して行われる。ユーザは学内だけでなく、学外からも受講、成績報告を行う可能性があるため、受講情報確認システム自体は学外公開する必要がある。このため、成績確認システム側でのセキュリティ対策に十分注意を払う必要がある。

● ユーザの成績報告

各ユーザにとってみれば、ユーザが成績報告するという手間が増えることとなる。この点は確認方式(1)、(2)では発生しないように見えるが、受講したユーザに対してVPN等のサービスを提供するような場合に、受講したかどうかをユーザに申告してもらわないと、管理者が気づかないため、ユーザの手間としては変わらない場合もある。

以上の問題点を表1にまとめて記載した。

3. 次期受講情報確認システムの実装

前節までで述べたように、認証に関連する問題点を整理すると、方式(3)は受講情報システムの管理者負担やIdP管理者との交渉が不要な点など利点が多く実装に値する。システムを学外公開する必要がある点が問題だが、申請を学内に限定するなど、各機関のセキュリティ状況に応じて対応できる。このため、受講情報確認システムに方式(3)を採用し、実装することとしたい。本節では実装に向けた仕様の詳細を記載する。

前節でも述べたように、ユーザのブラウザを介して学認連携 Moodle から受講情報確認システムへの登録を実現する。つまり、プラグイン動作をより詳細に述べると下記の

通りとなる。

- (1) ユーザは Moodle 上の成績確認ボタンを押す。
- (2) Moodle プラグインは、Moodle 上の DB を検索して、ユーザの成績を取得し、ユーザの成績、ePTID を、javascript により自動 POST を行う HTML ページを返す。
- (3) ブラウザは、受講情報確認システムに POST する。(2)の過程で、プラグインは下記のようなページをウェブブラウザに返す。このページは、javascript を利用して、読み込んだブラウザで受講情報確認システムに対してフォームを自動 POST するものである。

```
<body onload="document.forms[0].submit()">
<form method="POST" action="受講情報確認システム">
<input type="hidden" name="ePTID" value="ePTID">
<input type="hidden" name="score" value="成績">
</form></body>
```

受講情報確認システムでは、POST されたパラメータを見て、ePTID と成績を取得することができる。また、受講情報確認システムにアクセスするためには、ユーザがログインする必要があるため、ログイン ID もここで紐づけることができる。このようにして、ブラウザを経由して、学認連携 Moodle から、受講情報確認システムへの情報の受け渡しを行う。

実際には、学認連携 Moodle も受講情報確認システムも、各大学の IdP で認証される。IdP での認証プロセスも含めて、一連の認証と情報のやりとりは下記の通りとなる。

- (1) ユーザが学認連携 Moodle にログインする
- (2) 学認連携 Moodle の成績報告ボタンを押す
- (3) Moodle プラグインがウェブブラウザに ePTID と成績を POST するページを返す
- (4) ブラウザは受講情報確認システムに ePTID と成績を POST する
- (5) ユーザは受講情報確認システムにログインしていないので、IdP に SAML 認証要求を POST するページをブラウザに返す。
- (6) ブラウザは、IdP に SAML 認証要求を POST する。
- (7) IdP でユーザは既に認証済みなので、IdP は SAML 認証レスポンスを POST するページをブラウザに返す。
- (8) ブラウザは SAML 認証レスポンスを受講情報確認システムに POST する。
- (9) 受講情報確認システムで認可され、最初アクセスしたページにリダイレクト。
- (10) ブラウザは受講情報確認システムに ePTID と成績を POST する
- (11) 受講情報確認システムは ePTID と成績を受信し、IdP から受け取ったユーザ ID と紐づける。

	方式 (1)	方式 (2)	方式 (3)
受講情報確認システムの管理者負担	大きい	大きい場合あり	小さい
成績のリアルタイム反映	できない	定期実行間隔分の遅延あり	ユーザ報告次第即時
IdP 管理者との交渉	必要	必要	不要
ユーザによる送信確認	できない	できない	成績送信時に可能
受講情報確認システムの学外公開	不要	学認連携 Moodle に対して必要な場合あり	必要

表 1 確認方式と問題点の比較表

Table 1 Comparison table between the confirmation methods and problems of the methods

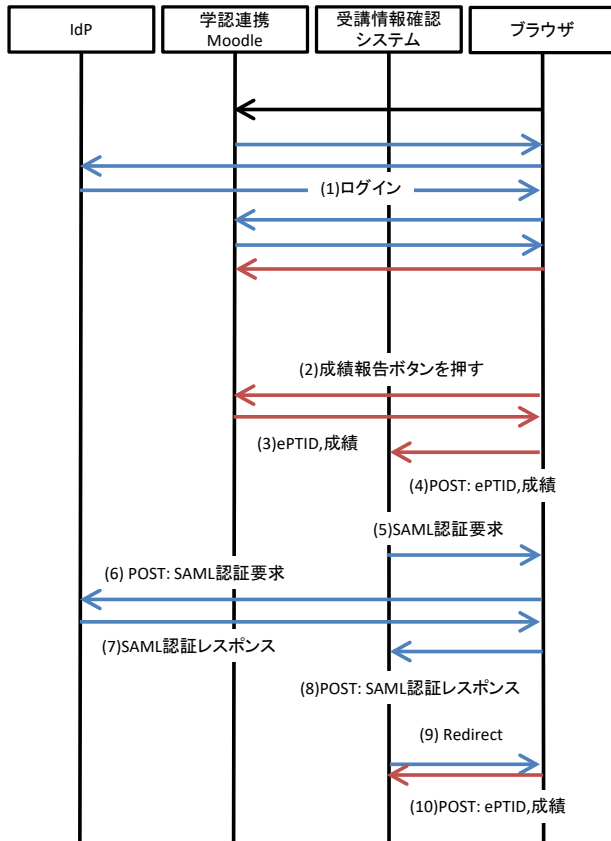


図 5 学認連携 Moodle 及び受講情報確認システムの認証の概要図
Fig. 5 Overview of the authentication of the Gaku-Nin moodle and the cyberethics grade confirmation system

3.1 検証内容と結果

我々は、この認証の流れを検証するため、群馬大学の学内環境を利用して、検証をおこなった。学認連携 Moodle の代わりとして、群馬大学 IdP のローカル SP として稼働している群馬大学 Moodle を利用した。受講情報確認システムも、群馬大学 IdP のローカル SP として稼働している。学認連携 Moodle は NII の DS を介して公開されているが、群馬大学 Moodle は公開されていない点、実際と異なるが Shibboleth 認証による SSO であり、本質的に違うことはない。

本検証では、群馬大学 Moodle 上に、成績を報告するた

めの php スクリプトへの単純なリンクを作成した。本来であれば、Moodle プラグインを作成し、Moodle と一体化したユーザインタフェースを提供するべきだが、データの流れを検証するためには必要ないので、本検証では単純化した。

検証に利用した php スクリプトは、ePTID、成績の情報を受講情報確認システムに渡すことに成功した。特に、(9)、(10)の段階で、Redirect に関して、POST されるかどうか自明ではなかったが、検証の結果、正常に POST リクエストを発行することを確認した。

4. まとめ

本論文では、受講情報確認システムの概要と、その成績確認の方法の実装について述べた。複数の成績確認の方法について比較を行い、成績確認のリアルタイム反映、受講情報管理システムの管理者負担、ユーザによる属性確認などの点に対して、確認方式 (3) が有利になることを示した。確認方式 (3) では、ブラウザを介して成績情報の送信を行うが、その実装について、Shibboleth IdP と組み合わせた環境で検証を行った結果を述べた。

今後は、実際に学認連携 Moodle で運用可能な実装を行っていく。学認連携 Moodle、受講情報確認システム共に、Shibboleth の枠組みからすると SP となるが、学認連携 Moodle から受講情報確認システムに向かって、成績というユーザの属性情報を渡すこととなっている。本来 SP の学認連携 Moodle が成績属性を提供する IdP として振る舞うことになる。通常の IdP と SP の属性のやり取りは、SAML の枠組みでアサーションのやり取りをして実現しているが、今回の検証では、単純に属性値そのものをやり取りした。Shibboleth の枠組みでは、メタデータによりお互いが公開鍵を有することで、アサーションの電子署名を行い、アサーションの正当性の検証を行っているが、今回は署名の検証までは実現していない。今後は、属性のやり取りの SAML 化や署名検証が必要かどうかを検討し、成績確認に最適な方法で実装を行う予定である。

謝辞 本研究は JSPS 科研費 JP17K00474 の助成を受けたものです。

参考文献

- [1] 国立情報学研究所, 高等教育機関の情報セキュリティ対策のためのサンプル規程集, <http://www.nii.ac.jp/service/sp/> (2018年2月4日現在)
- [2] 上田浩, 中村素典, 古村隆明, 神智也: 倫倫姫プロジェクト-学認連携 Moodle による多言語情報倫理 e ラーニング, デジタルプラクティス, 6, 2, 97-104, 2015.
- [3] 上田浩: 倫倫姫プロジェクト: 多言語情報倫理 e ラーニングコンテンツの開発と運用, 大学 ICT 推進協議会 2011 年度年次大会論文集, 557-561 (2011)
- [4] 浜元 信州, 上田 浩: 「学認連携 Moodle 講習成績確認システムの開発と運用」, 学術情報処理研究」学術情報処理研究 Vol 20 p21 (2016)
- [5] 浜元信州, 久米原栄, 上田浩: 「群馬大学での学認連携 Moodle 講習サイトの利用について」大学 ICT 推進協議会 2015 年次大会
- [6] 浜元信州, 横山重俊, 上田浩: 「学認連携 Moodle 受講情報確認システムでの管理権限の実装」大学 ICT 推進協議会 2017 年次大会
- [7] 受講情報確認システム, <https://github.com/n-hamamoto/cegc> (2018年2月4日現在)
- [8] Shibboleth 認証, <https://www.switch.ch/aai/demo> (2018年2月11日現在)