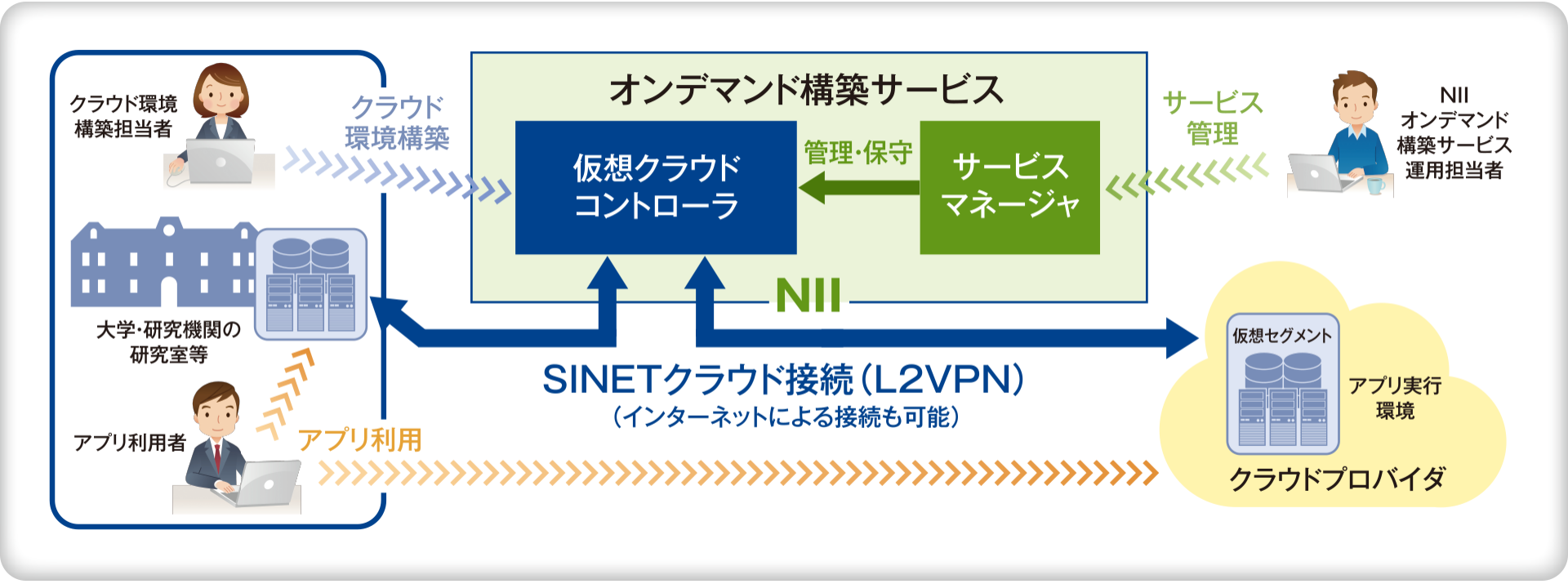
ハイブリッドクラウドでのログ解析環境の紹介

浜元信州
群馬大学総合情報メディアセンター

はじめに

- 学認クラウドオンデマンド構築サービス

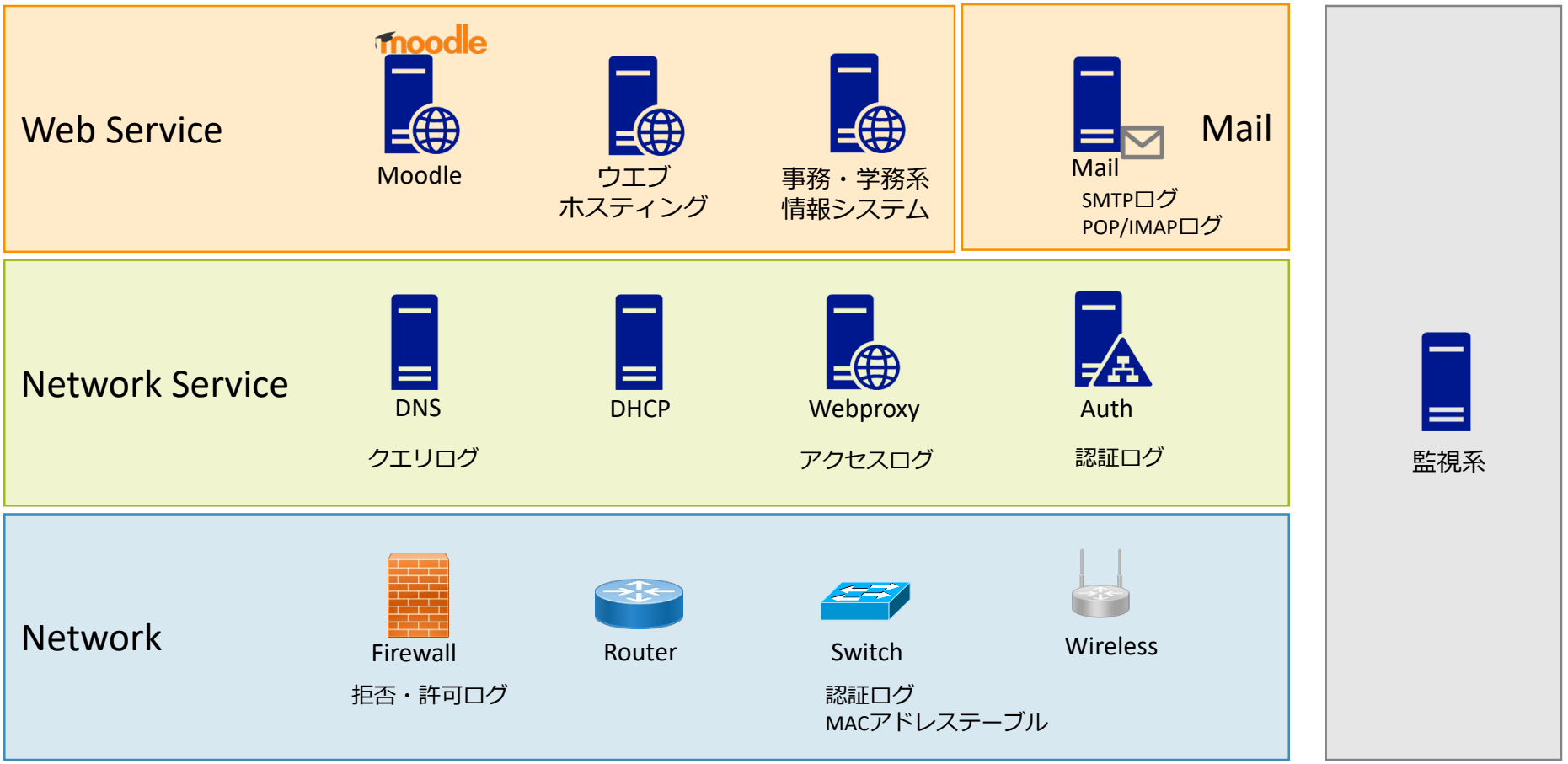


アプリケーション テンプレート

- Moodle
  - 前回のオープンフォーラムで発表
  - テンプレート改善中！
  - 今回のハンズオンで最新版を紹介します！

- ログ解析環境（予定）
  - 現在開発中
  - 本発表で適用例を紹介します。

学内ログ


Web Service
moodle
Moodle
ウエブ
ホスティング
事務・学務系
情報システム
Mail
Mail
SMTPログ
POP/IMAPログ
Network Service
DNS
クエリログ
DHCP
Webproxy
アクセスログ
Auth
認証ログ
監視系
Network
Firewall
拒否・許可ログ
Router
Switch
認証ログ
MACアドレステーブル
Wireless

ログ解析

- 学内にある様々なサービスのログを解析して，有用な結果を得たい。

  - セキュリティ対策
    - セキュリティインシデント調査
    - **端末特定**
    - インシデント発見

  - ウエブサービス運用
    - **サーバ資源最適化**
    - ラーニングアナリティクス

端末特定のための
ログ解析環境

背景と目的

- 高度化したサイバー攻撃への対策が必要とされている

- 高度化した攻撃の特徴
  - 影響の深刻化：機密情報の漏洩，サービス停止
  - 手口の巧妙化：標的型攻撃
  - 期間の長期化：長期間の潜伏

- 対策
  - 入口対策（これまで）：公開サーバ制御，ポート閉塞
  - 出口対策（これから）：感染してからの調査

（大学での）
標的型攻撃対策に必要なこと

- 内部での感染活動等の発見（出口対策）
  - 各大学でのログ解析の高度化

- 外部機関からの脅威情報の入手・共有
  - NII-SOCS，警察，外部SOCサービスなど

→インシデント通告時の速やかな対応
→機関内部のログと照合
→検知精度の向上

（大学での）
標的型攻撃対策に必要なこと

- ログ分析
  - 外部SOCや通報に対応して，速やかに端末を特定する
  - 内部で発生しているインシデントを特定する
    - 発見手法によっては計算資源が必要
- ログアーカイブ
  - ため込む期間（3年？）が長いとストレージが不足
- 手順の汎用化
  - 多くの機関で利用可能であること

ログ解析環境とクラウド

- ログ分析
  - 発見手法によっては計算資源が必要
- ログアーカイブ
  - ため込む期間（3年？）が長いとストレージが不足
- 手順の汎用化
  - 多くの機関で利用可能であること.

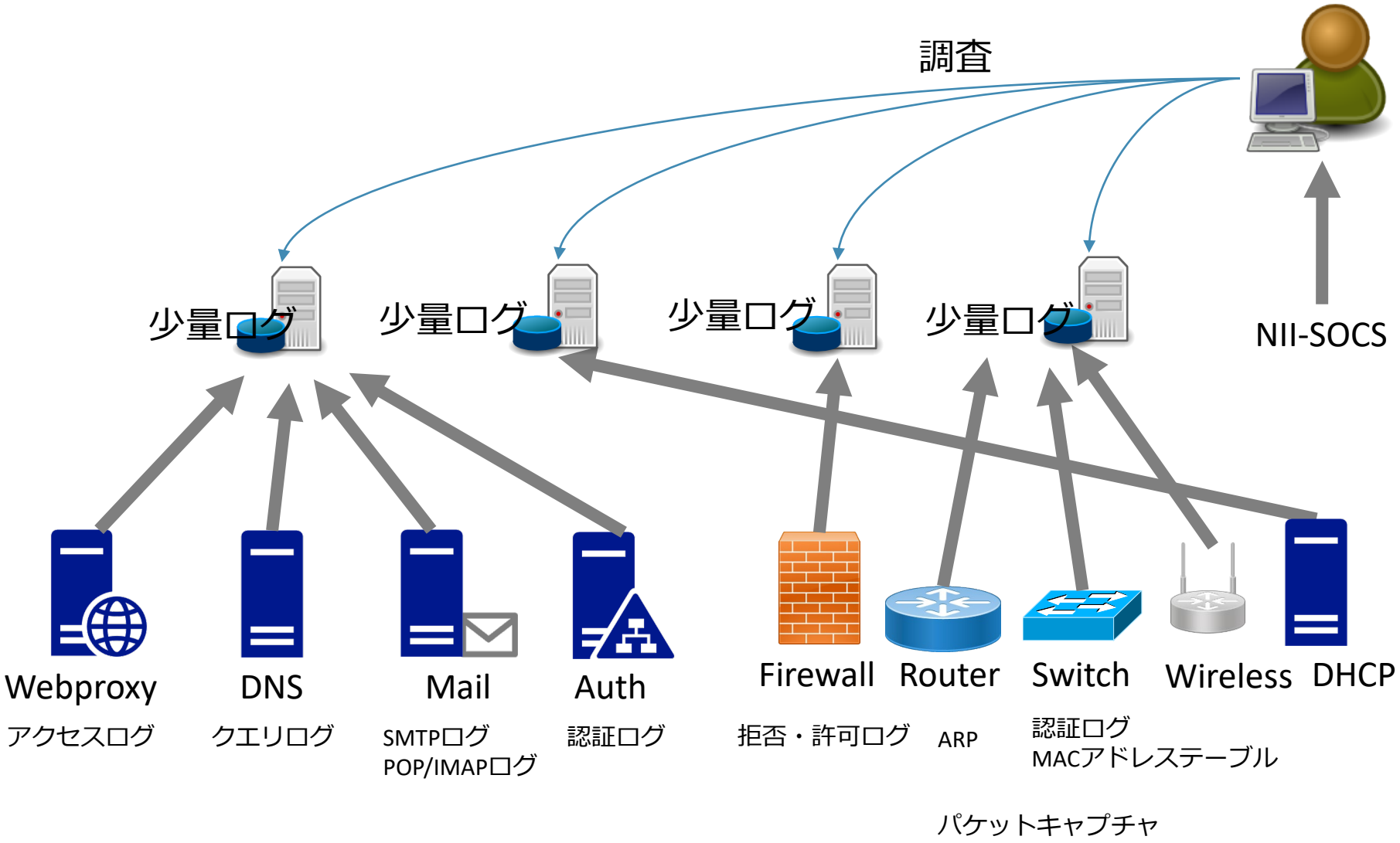
- クラウドの特徴

→
- 計算資源確保が容易

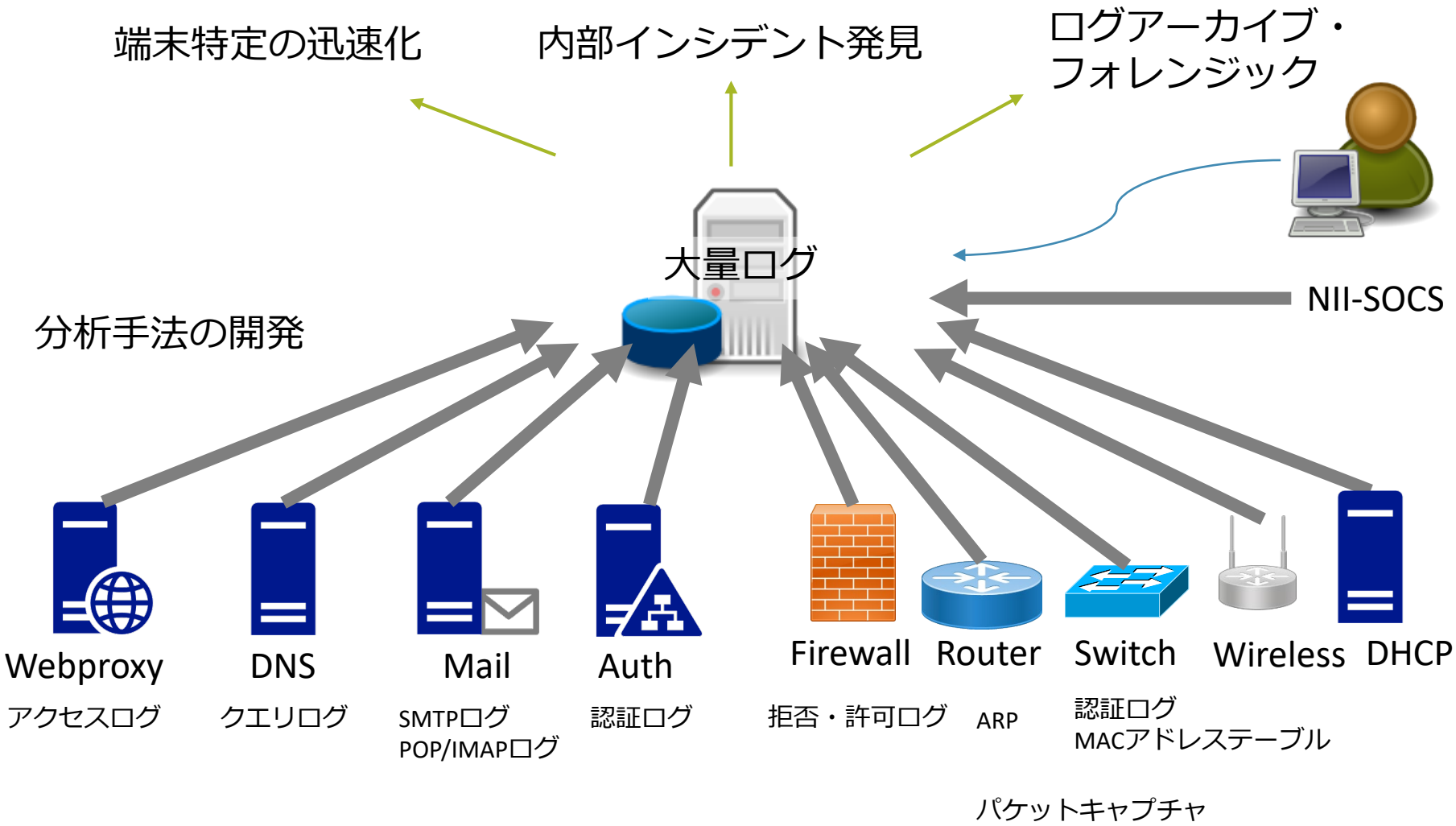
- オブジェクトストレージ
→
- コールドストレージ

→
- テンプレート化

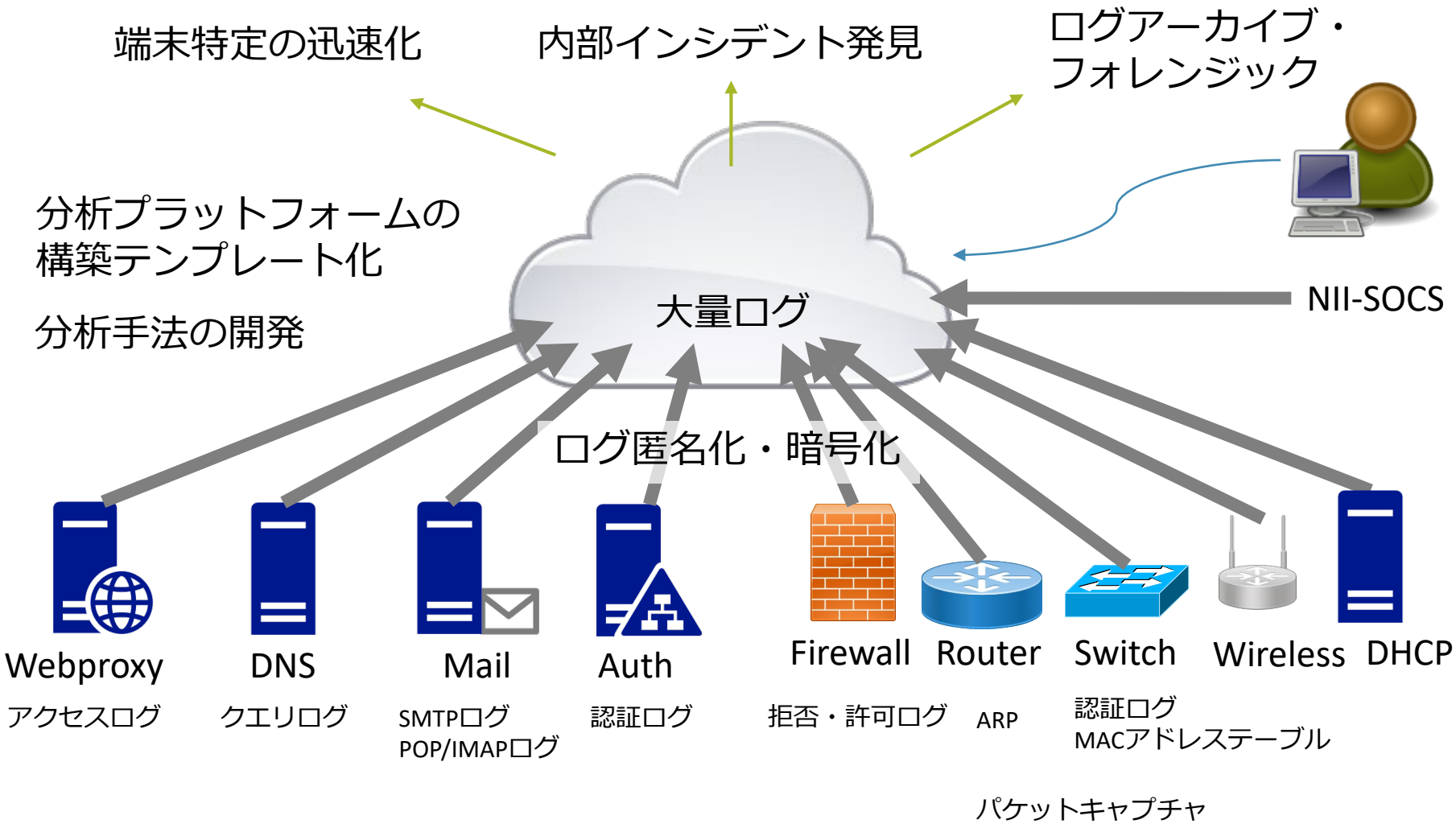
ログ解析環境（現状）


調査
少量ログ
少量ログ
少量ログ
少量ログ
NII-SOCS
Webproxy
アクセスログ
DNS
クエリログ
Mail
SMTPログ
POP/IMAPログ
Auth
認証ログ
Firewall
拒否・許可ログ
Router
ARP
Switch
認証ログ
MACアドレステーブル
Wireless
DHCP
パケットキャプチャ

ログ解析環境


端末特定の迅速化
内部インシデント発見
ログアーカイブ・
フォレンジック
大量ログ
分析手法の開発
NII-SOCS
Webproxy
アクセスログ
DNS
クエリログ
Mail
SMTPログ
POP/IMAPログ
Auth
認証ログ
Firewall
拒否・許可ログ
Router
ARP
Switch
認証ログ
MACアドレステーブル
Wireless
DHCP
パケットキャプチャ

クラウド上のログ解析環境



ログ解析環境の要件（クラウド）

- 要件
  - クラウド上へ機密情報を保存する際には，十分な情報漏洩対策を施すこと.

→オンプレミスとクラウドの適切な使い分け
必要に応じた暗号化，匿名化

今回のターゲット

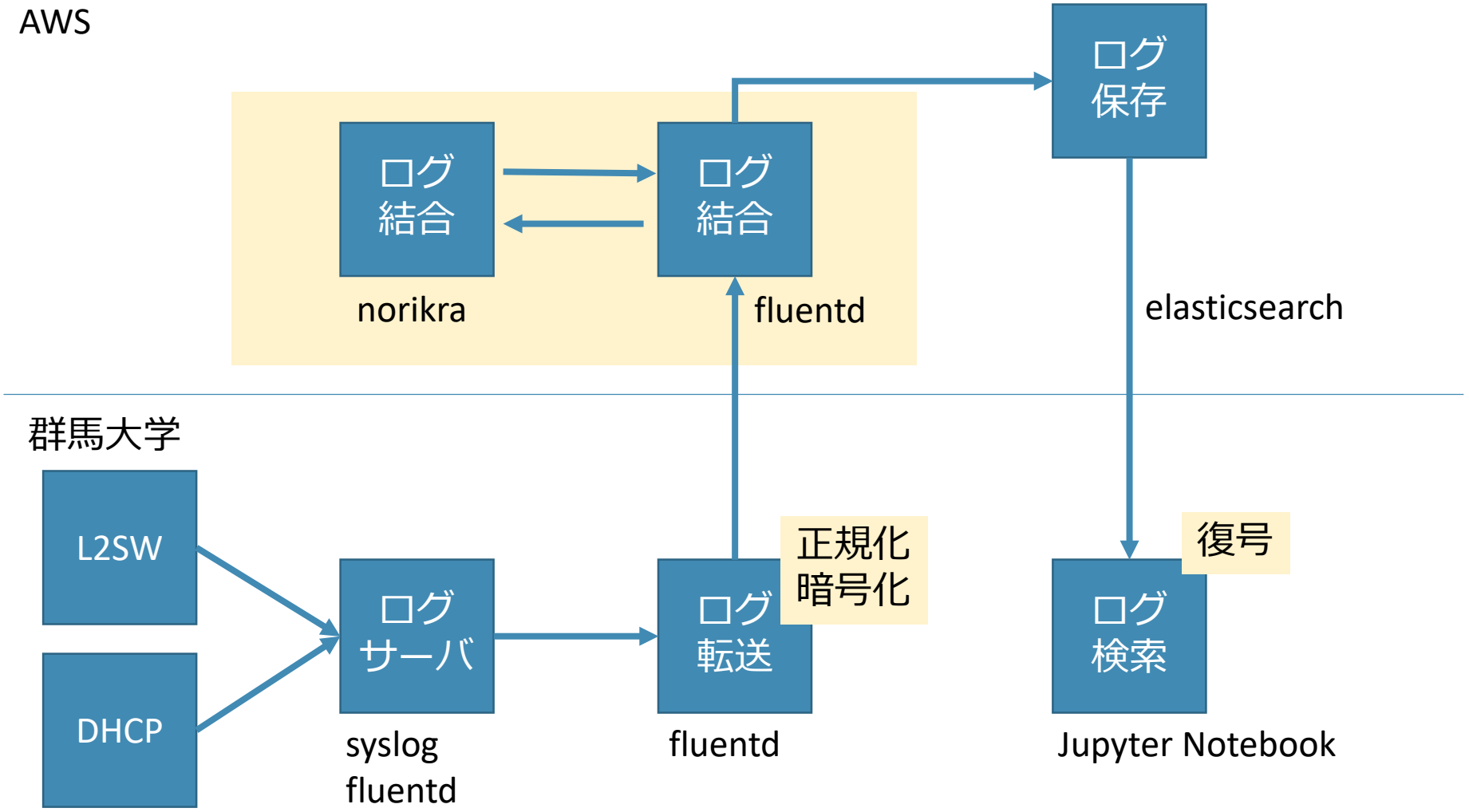
- 外部SOCや通報に対応して，速やかに端末を特定する

- 内部で発生しているインシデントを特定する
  - 発見手法によっては計算資源が必要

- 過去ログをアーカイブする
  - ため込む期間（3年？）が長いとストレージが不足

- オンプレミスで資源が不足するならクラウドを利用する

今回構築したログ解析環境



端末特定

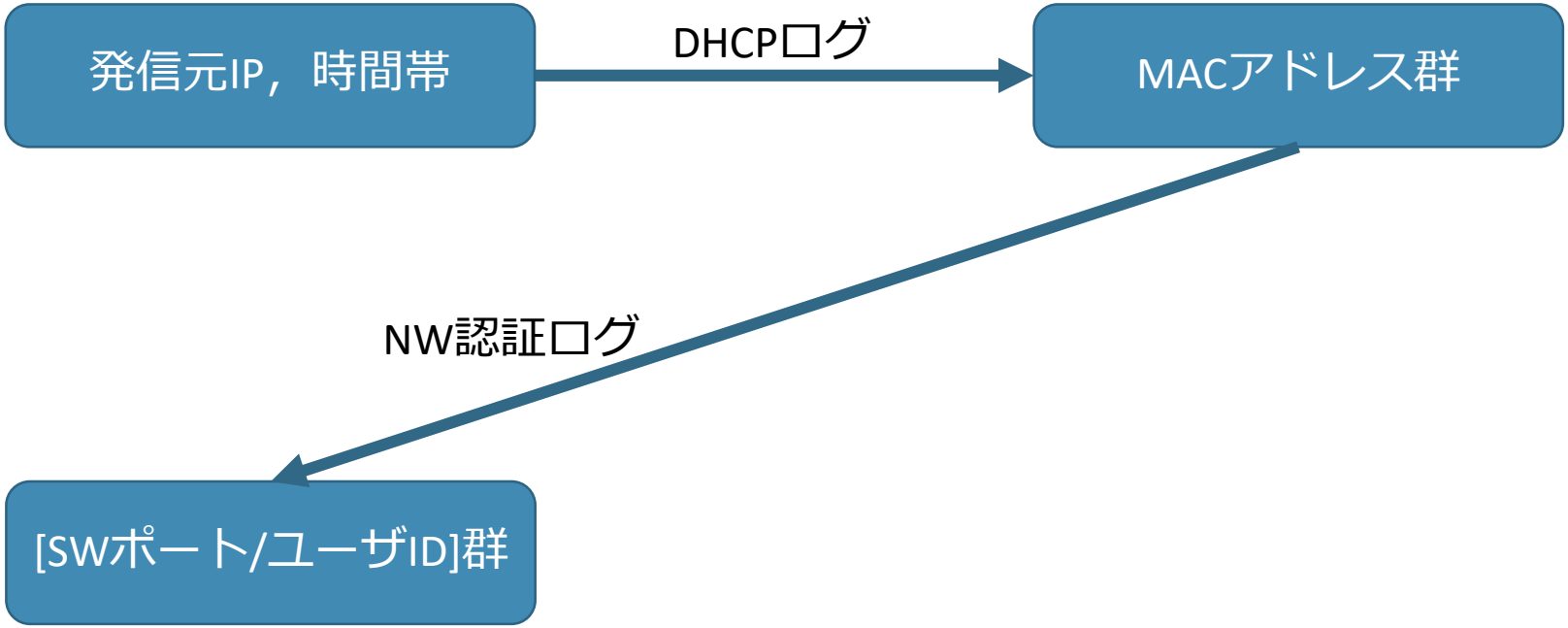
- SOCなどからの情報
  - 発信元IPアドレス，時間帯
  - 宛先IPアドレス，宛先URL

- 想定するネットワーク環境
  - 固定IP・グローバルIP
  - **動的IP・グローバルIP**
  - NAPT環境・プライベートIP
  - IPv6環境

端末特定（今回想定）

DHCP環境

DHCPログ：ICS DHCPd
NW認証ログ：ALAXLA製スイッチのMAC/ユーザ認証



ログ解析環境

- ログ転送（学内→AWS）：
  - ログサーバからのログを受けて，必要部分の暗号化・匿名化処理をし，AWSに転送する。
- ログ結合：
  - L2SWとDHCPのログを結合しておく
- ログ検索：
  - 暗号化を解除して検索結果を表示する。

暗号化

- DATA masking filterを作成

  - deletion　　消去する
  - masking-out　　xでマスクする
  - encryption　　暗号化する
  - substitution　　ハッシュ化する
  - substitution-ip　　サブネットを維持してIP置換

正規化

- Normalize Filterを作成
  - Mac AddressのNormalizeを行う。
    以下の形式のMACアドレスは１つに統一

  - xxxx.xxxx.xxxx
  - XX:XX:XX:XX:XX:XX
  - XX-XX-XX-XX-XX-XX

結合

- DHCPのログとL2SW認証ログを結合

DHCPログ

2016 Jun 22 03:17:35 ara-dhcp02 DHCPACK on 192.168.219.187 to **c0:e0:28:3c:52:07** (pc_name_R5jG7mnr9sWNXFhhU2Hazg) via eth0.7

L2SWログ

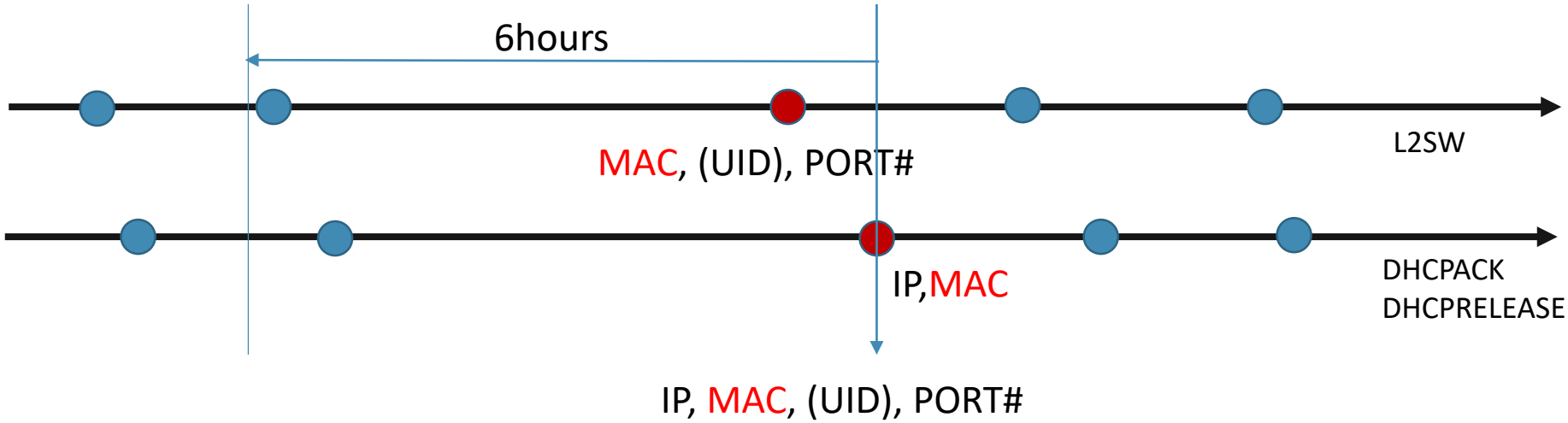
2016 Jun 22 05:29:35 aoa00011 AUT 06/22 05:29:35 MAC No=1:NORMAL:LOGIN: MAC=**c0e0.283c.5207** PORT=0/14 VLAN=219 Login succeeded.

統合ログ

2016 Jun 22 03:17:35 ara-dhcp02 DHCPACK on 192.168.219.187 to **c0:e0:28:3c:52:07** (pc_name_R5jG7mnr9sWNXFhhU2Hazg) via eth0.7 2016 Jun 22 05:29:35 aoa00011 AUT 06/22 05:29:35 MAC No=1:NORMAL:LOGIN: MAC=**c0e0.283c.5207** PORT=0/14 VLAN=219 Login succeeded.

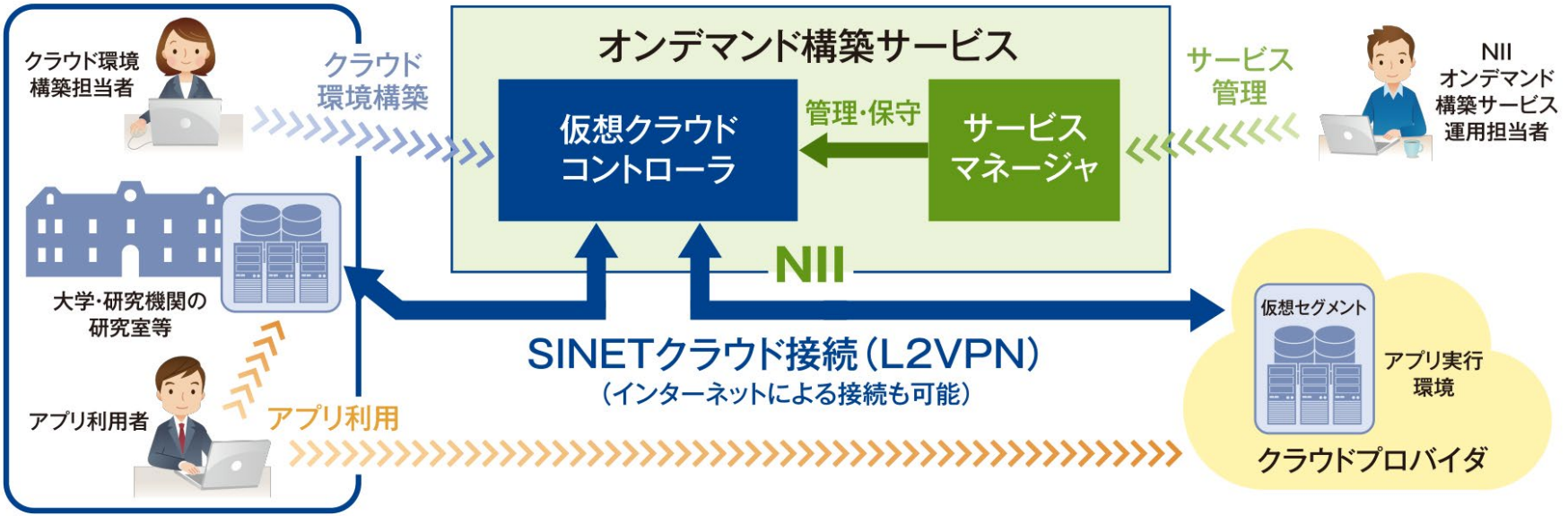
Norikra EPL

```
SELECT
   d.message as dhcp_message, d.timestamp as timestamp,
   l.message as l2sw_message, l.timestamp as l2sw_timestamp,
   d.dhcp_ipv4 as ipv4, d.dhcp_mac as mac
FROM dhcp.std:unique(dhcp_mac).win:time(6 hours) as d, l2sw.win:time_batch(60 sec) as
l
WHERE
   d.dhcp_mac = l.l2sw_mac
   AND ( d.dhcp_msg = 'DHCPACK' OR d.dhcp_msg = 'DHCPRELEASE' )
```

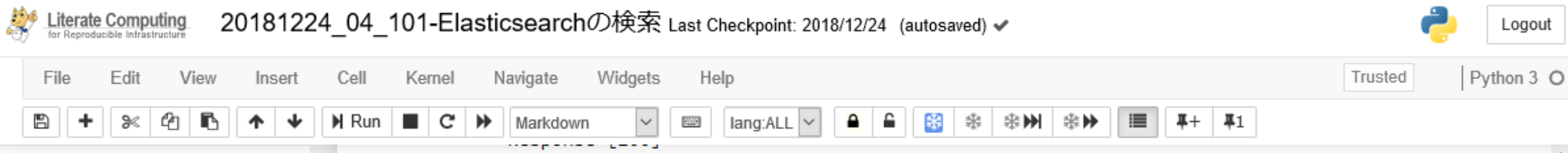


実装方法

オンデマンド構築サービス



端末特定Notebook Sample



Contents

4 検索条件の入力

検索対象となるIPアドレスを指定してください。

[154]:
```
# (例)
# target_ip = '192.168.56.231'

#target_ip ='192.168.12.173'
target_ip ='192.168.133.96'
#target_ip ='192.168.95.47'
```

検索対象の基準となる日時を指定してください。なお、日時は日本時(JST)で入力されたものとみなします。

[155]:
```
# (例)
# target_time = '2018-11-06 07:20:00'

target_time ='2019-01-04 03:46:00'
```

基準となる日時からどの範囲までを検索対象とするかを指定します。指定方法の詳細についてはtimedeltaを参照してください。

[156]:
```
import datetime
# (例)
# search_range = datetime.timedelta(hours=1)  # 前後１時間を指定する場合
# search_range = datetime.timedelta(minutes=30)  # 前後30分を指定する場合

search_range =datetime.timedelta(hours=24)
```

ログの検索クエリ

```
es_search = {
    'size': 10,
    'sort': ['timestamp'],
    'query': {
        'bool': {
            'filter': [
                {'term': {'ipv4': dm_encode(target_ip, password, salt)}},
                timestamp_range,
            ]
        }
    },
}
```

Contents

5.3 検索の実行

検索を実行します。

[163]:
```
import requests
r = requests.get(
    es_url,
    json=es_search,
)
print(r)
```

```
<Response [200]>
```

検索条件に合致したログを表示します。

[164]:
```
for x in r.json()['hits']['hits']:
    src = x['_source']
    ts = datetime.datetime.strptime(
            src['timestamp'], '%Y-%m-%dT%H:%M:%S%z')
    ts2 = datetime.datetime.strptime(
            src['l2sw_timestamp'], '%Y-%m-%dT%H:%M:%S%z')
    print('{0} {2}\n{1} {3}\n'.format(
        ts.astimezone(JST).strftime('%Y-%m-%d %H:%M:%S'),
        ts2.astimezone(JST).strftime('%Y-%m-%d %H:%M:%S'),
        dm_decode(src['dhcp_message'], password, salt),
        dm_decode(src['l2sw_message'], password, salt)
    ))
```

```
2019-01-04 13:10:40 DHCPACK on 192.168.133.96 to 8a:c0:04:e4:11:85 (pc_name_DCWDUVYuBoEat
/kEuzbBpQ) via eth0.32
2019-01-04 13:11:39 AUT 06/22 03:45:53 MAC No=266:NORMAL:SYSTEM: MAC=8ac0.04e4.1185 Resta
rt authenticating for MAC address.

2019-01-04 13:10:40 DHCPACK on 192.168.133.96 to 8a:c0:04:e4:11:85 (pc_name_DCWDUVYuBoEat
/kEuzbBpQ) via eth0.32
2019-01-04 13:11:39 AUT 06/22 03:45:53 MAC No=256:NORMAL:LOGIN: MAC=8ac0.04e4.1185 PORT=0
/4 VLAN=133 Reauthentication succeeded.
```

クラウド上のログ

[168]:
```
for x in r.json()['hits']['hits']:
    src = x['_source']
    ts = datetime.datetime.strptime(
            src['timestamp'], '%Y-%m-%dT%H:%M:%S%z')
    ts2 = datetime.datetime.strptime(
            src['l2sw_timestamp'], '%Y-%m-%dT%H:%M:%S%z')
    print('{0} {2}\n{1}\n'.format(
        ts.astimezone(JST).strftime('%Y-%m-%d %H:%M:%S'),
        ts2.astimezone(JST).strftime('%Y-%m-%d %H:%M:%S'),
        src
    ))
```

```
2019-01-04 13:10:40 {'ipv4': '_aKZ0xPeVlfAdBIJlJh45w', 'dhcp_message': 'z4nBxM3OQhIIzixIX
ur068UoSTEhFy8xbVunSAEx_kg5WKWsN5U8n2ze-zU7FYwkXRfUKMrAaoeZMKkAs4mRSPkhPfW20IV3kMjdGtdvIL
NdTDYd3e3Mw4pQJi5KY5so', 'l2sw_message': 'Bx5YJoxsWaoOFZOVHyD-7zwl6qiL2R-bO7QX51_DgzXivIo
Zb811c8Ivm7BlhyTrp_18QaL9a6SKZIh6_Nre886fLnYJqSueNey7oLY47uBNRqLDo_6EM8lC31gG6BECwWBpzeVA
xZCgBTqF_mlrvw', 'mac': 'GVAyzr7IKLJS47dIv8uNLz5vodUBCUw6h6s6MoFYA30', 'timestamp': '2019
-01-04T04:10:40+0000', 'l2sw_timestamp': '2019-01-04T04:11:39+0000'}
2019-01-04 13:11:39

2019-01-04 13:10:40 {'ipv4': '_aKZ0xPeVlfAdBIJlJh45w', 'dhcp_message': 'z4nBxM3OQhIIzixIX
ur068UoSTEhFy8xbVunSAEx_kg5WKWsN5U8n2ze-zU7FYwkXRfUKMrAaoeZMKkAs4mRSPkhPfW20IV3kMjdGtdvIL
NdTDYd3e3Mw4pQJi5KY5so', 'l2sw_message': 'Bx5YJoxsWaoOFZOVHyD-7-AluektcY0PZE3jCIDQTlRdEMA
EfZzd4Y29LUfNncXtARHWmv0D0kDUHn68g0XlSmc48UHYo6ZBeVeFlpzI-dsNKq3Q3nBOeaw3AzwoE1C4RutKrMzr
yvSkrXoBASUTfw', 'mac': 'GVAyzr7IKLJS47dIv8uNLz5vodUBCUw6h6s6MoFYA30', 'timestamp': '2019
-01-04T04:10:40+0000', 'l2sw_timestamp': '2019-01-04T04:11:39+0000'}
2019-01-04 13:11:39
```

In [ ]:

クラウドを利用したログ解析環境のMoodleへの適用

Moodle運用

- サービスデスク
  - Moodle利用者からの質問受付窓口
- インシデント管理
  - 障害対応記録管理，利用
- 問題管理
  - **障害対応，予兆検知（今年度）**
- 構成管理
  - Moodle設定の構成情報の収集，管理，提供
- 変更管理
  - Moodleやサーバ基盤に行われた変更の許可，実装
- リリース管理
  - **テストと導入バージョンの管理（昨年度）**

Moodleに関連するログ

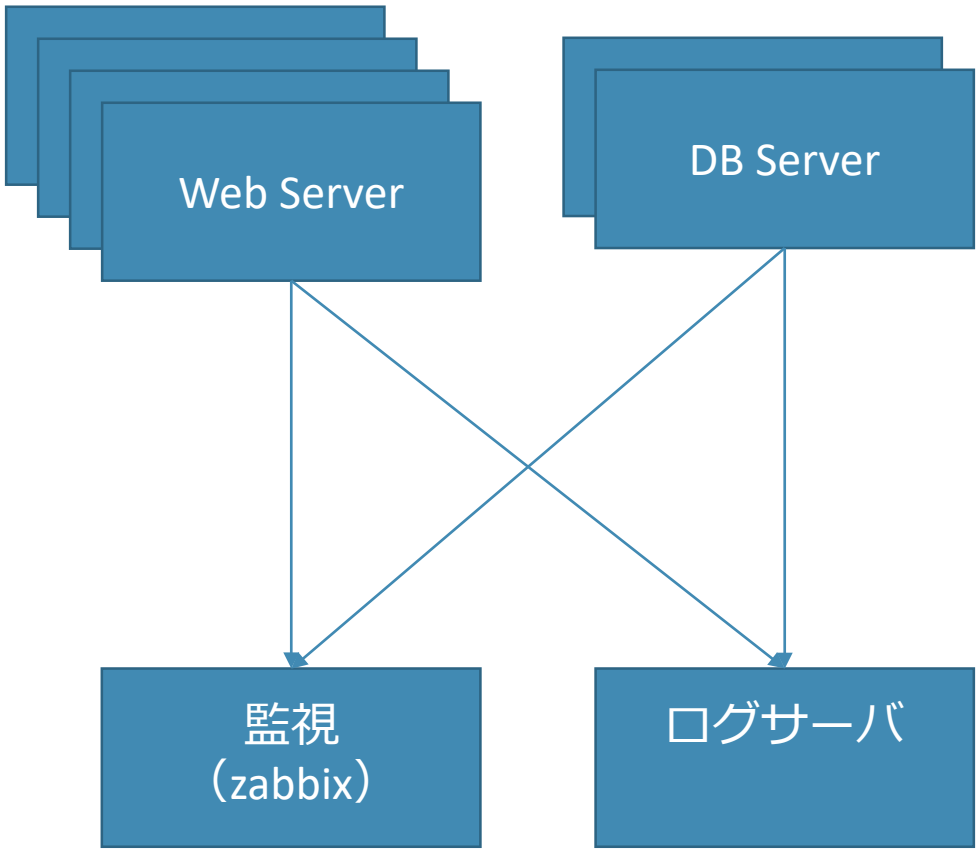
- Moodleソフトウエア（アプリケーション）
  - Moodle standard log：イベントログ
  - 時刻，フルネーム，イベント，対象，アクセスIP
- LAMP環境（プラットフォーム）
  - Web,PHP　　httpd/ssl_access.log, httpd/ssl_error.log
  - DB　　mysqld.log
  - OS　　/var/log/message
- 動作基盤（ハードウエア)
  - CPU，メモリ，HDD使用量
  - ネットワーク，HDDアクセス

Moodleのイベント

- 授業，自習用の教材配布
- 小テスト
- 課題提出
- 映像配信
- フォーラム利用
- フィードバック

- イベントログとサーバのログを関連させることで「小テスト見ていたら遅くなった」などの原因分析が可能になるかも

Moodleとログ管理



問題点：

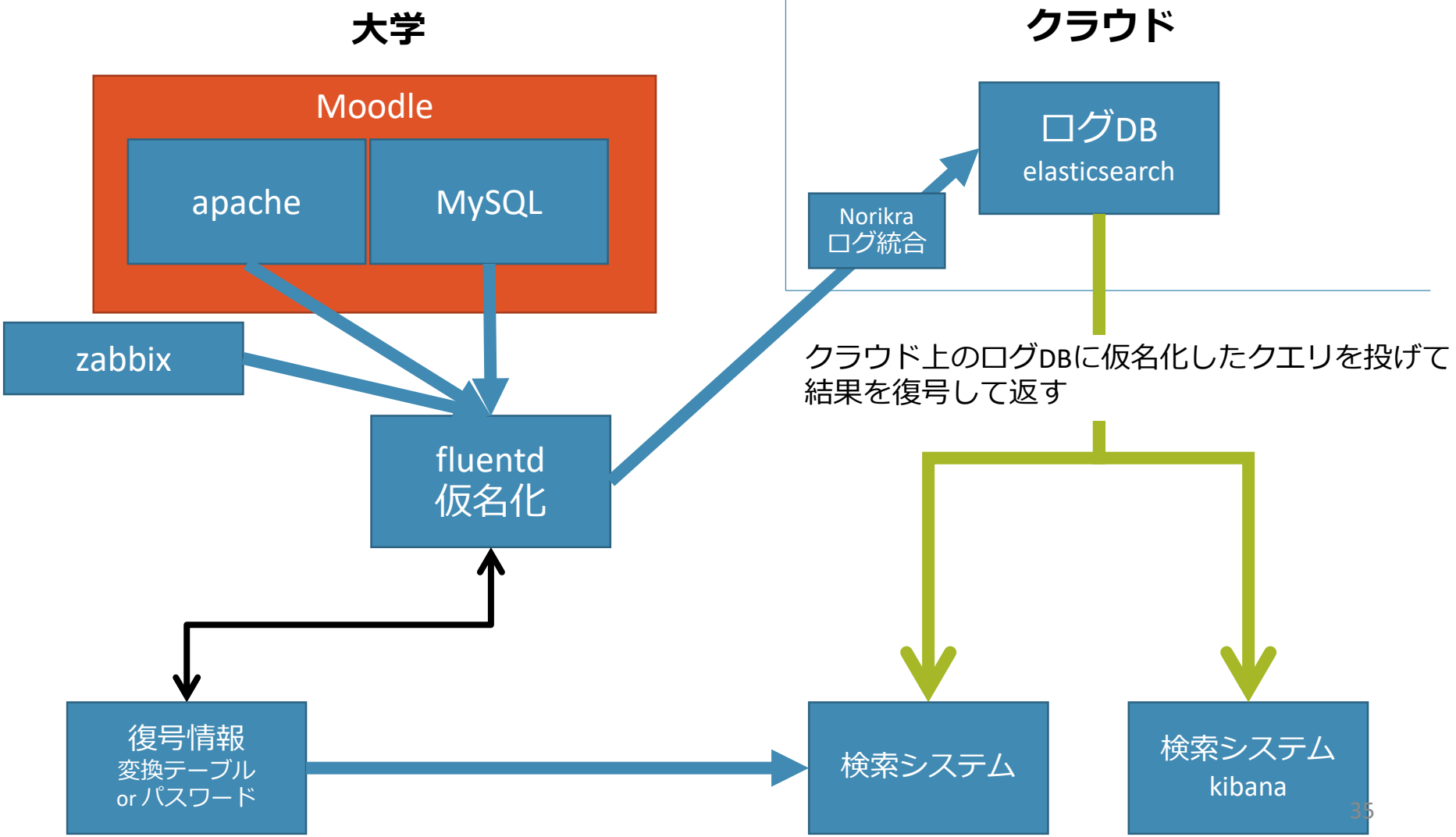
1. ログの量それなりにある。
   ログを蓄積する必要もある。
   保存領域の確保

2. ログが分散していて，相関分析が難しい

3. ログに個人情報がある

解決：
1クラウド利用で対応
2集約環境を提案
3暗号化で対応

ログ解析用クラウド環境（Moodleへの適用）



適用例

- 運用
  - 実運用で、ユーザが意識できる「行ったこと」（イベントログ）を手掛かりに原因追求できる
- サイジング
  - Jmeterなどのテストプランを用いた結果の解析

個人情報の暗号化

- フィールドを秘匿するプラグインを開発

下記のフィールドは暗号化してクラウドに保存

username,firstname,lastname,

email,middlename,

lastnamephonetic,

firstnamephonetic,

alternatename

```
<filter moodle.user.**>

   @type data_masking
   <field>
     key
username,firstname,lastname,email,middl
ename,lastnamephonetic,firstnamephonet
ic,alternatename
     type encryption
     password testpw
     salt 8zbuV1TZ
   </field>

</filter>
```

Moodle負荷試験への適用例

- 100ユーザのテストプラン
- コースサイズL（1GB）
- CPU2コア，メモリ16GB
- httpd最大200プロセス
  - 通常時100httpd（スペア100）その他86

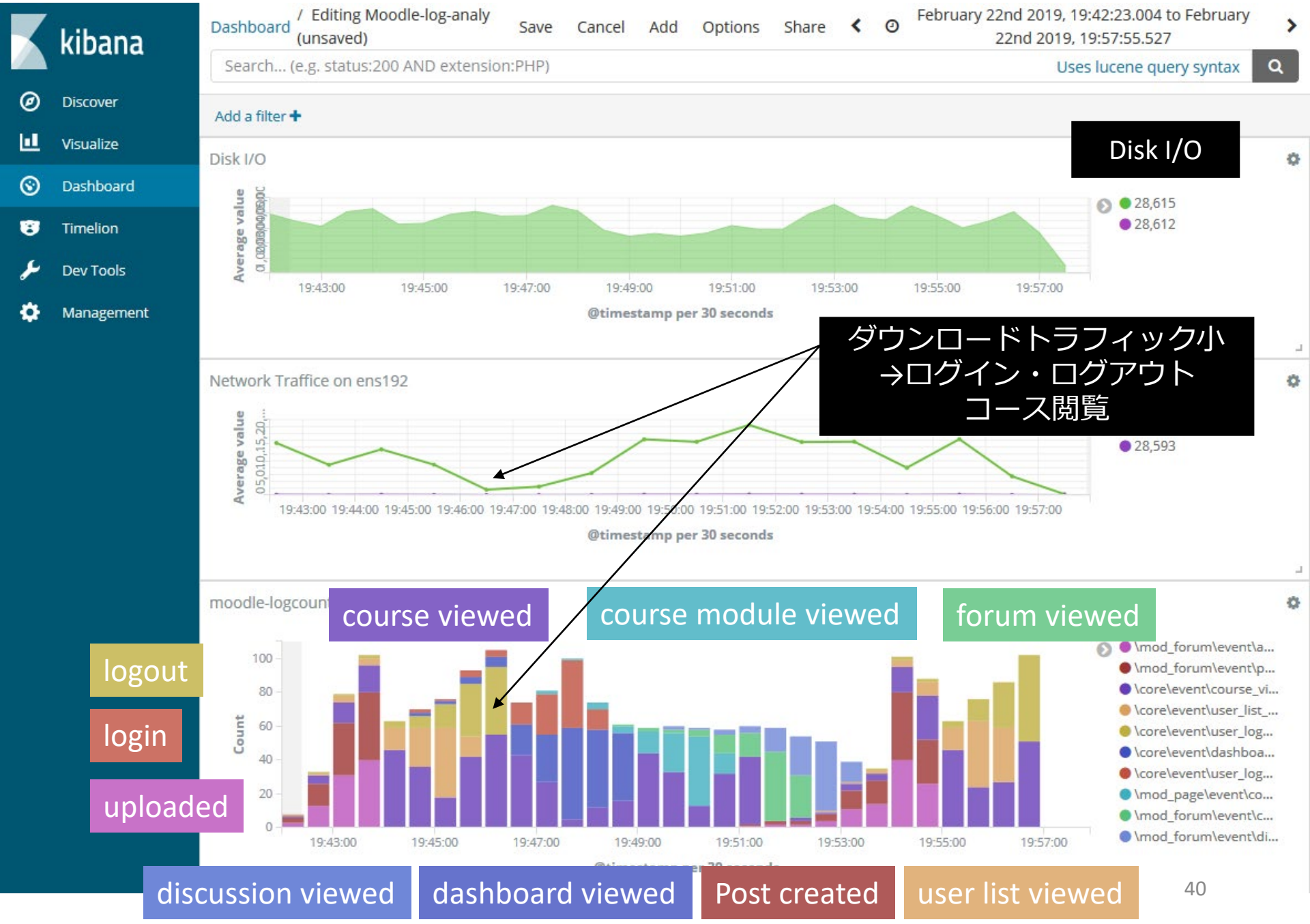
- スレッド数，Ramp-Up期間，ループ
  - スレッド100, Ramp-Up 40，ループ5
    - 100スレッドを40sかけて実行する
      （100人が40s間でアクセスを開始）

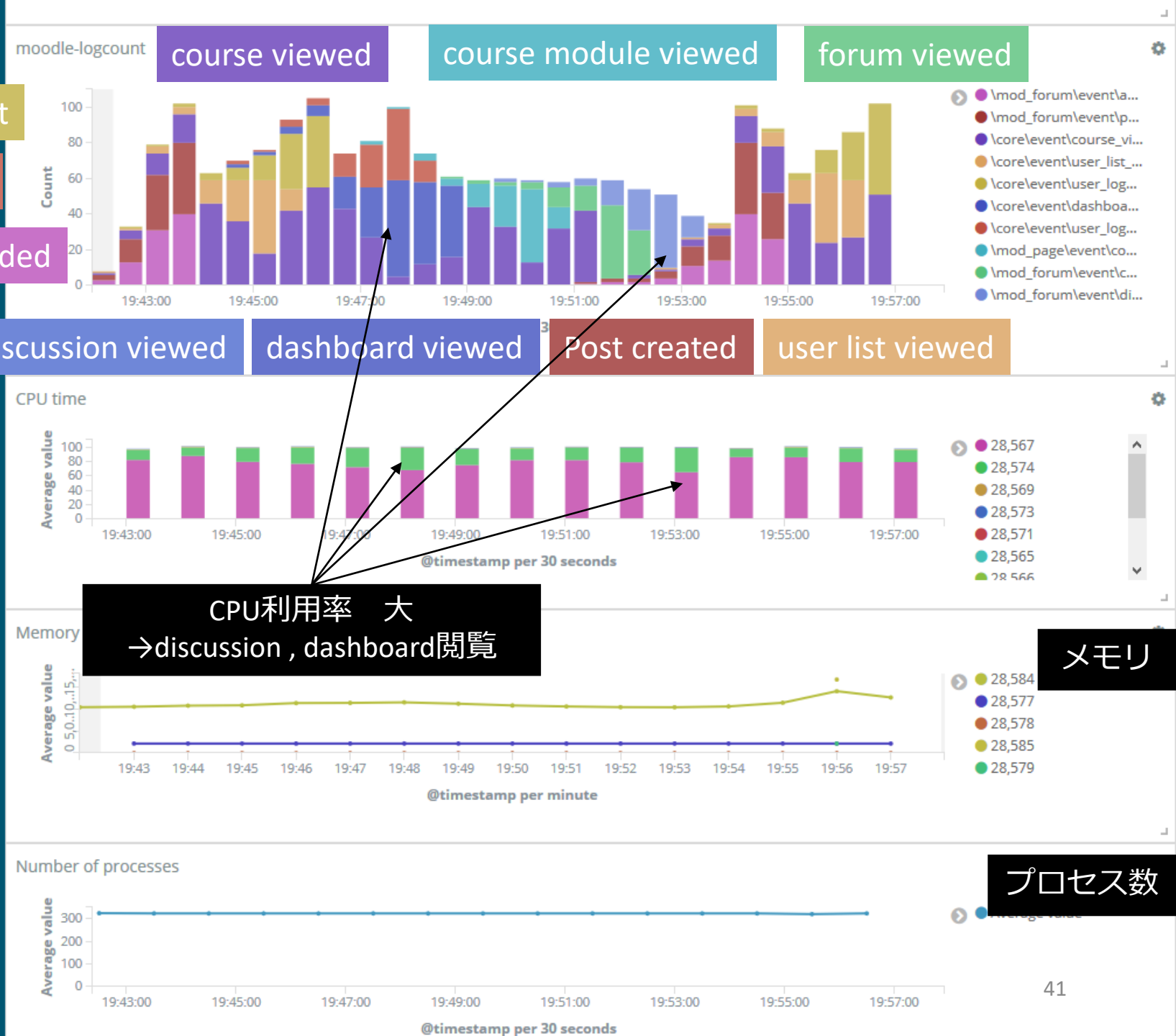
- ターゲットスループット
  - 120 サンプル数/分　に制限

Jmeterテストプラン

- ログイン
- ダッシュボードにアクセス
- コースにアクセス
- ログアウト

- ログイン
- コースにアクセス
- ページモジュールにアクセス
- フォーラムにアクセス
- ディスカッションを見る
- 返信する
- ユーザー一覧を見る
- ログアウト


kibana
Discover
Visualize
Dashboard
Timelion
Dev Tools
Management
Dashboard / Editing Moodle-log-analy (unsaved)
Save
Cancel
Add
Options
Share
February 22nd 2019, 19:42:23.004 to February 22nd 2019, 19:57:55.527
Search... (e.g. status:200 AND extension:PHP)
Uses lucene query syntax
Add a filter +
Disk I/O
Average value
28,615
28,612
@timestamp per 30 seconds
Network Traffice on ens192
28,593
ダウンロードトラフィック小
→ログイン・ログアウト
コース閲覧
moodle-logcount
course viewed
course module viewed
forum viewed
Count
logout
login
uploaded
discussion viewed
dashboard viewed
Post created
user list viewed


moodle-logcount
course viewed
course module viewed
forum viewed
logout
login
uploaded
discussion viewed
dashboard viewed
Post created
user list viewed
Count
\mod_forum\event\a...
\mod_forum\event\p...
\core\event\course_vi...
\core\event\user_list_...
\core\event\user_log...
\core\event\dashboa...
\core\event\user_log...
\mod_page\event\co...
\mod_forum\event\c...
\mod_forum\event\di...
CPU time
Average value
28,567
28,574
28,569
28,573
28,571
28,565
@timestamp per 30 seconds
CPU利用率　大
→discussion , dashboard閲覧
Memory
メモリ
28,584
28,577
28,578
28,585
28,579
@timestamp per minute
Number of processes
プロセス数
@timestamp per 30 seconds

course viewed

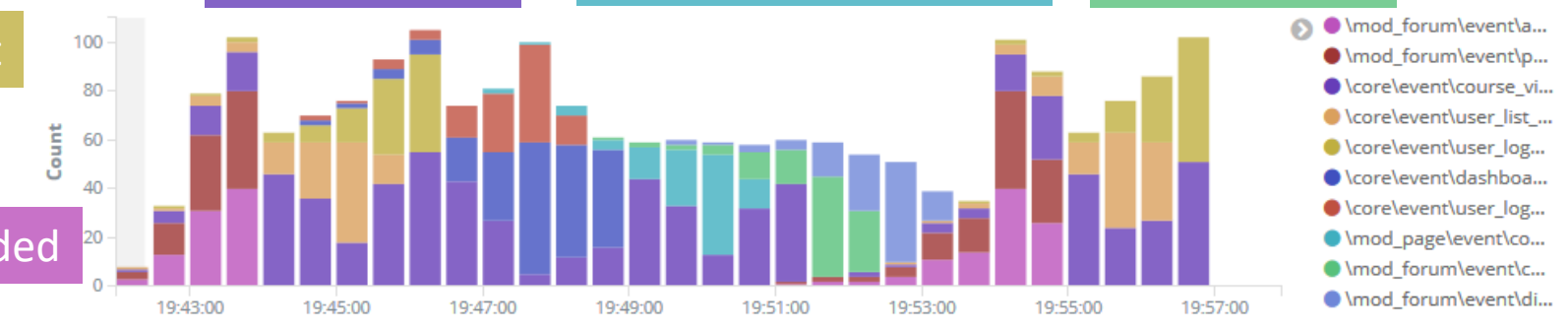
course module viewed

forum viewed

logout

login

uploaded



discussion viewed

dashboard viewed

Post created

user list viewed

moodle-count

1–50 of 2,069

Time	_source
February 22nd 2019, 19:56:55.000	id: 62,648 eventname: \core\event\user_loggedout component: core action: loggedout target: user object table: user objectid: 604 crud: r edulevel: 0 contextid: 1 contextlevel: 10 contextinstanceid: 0 userid: 604 courseid: 0 relateduserid: - anonymous: false other: a:1:{s:9:"sessionid";s:26:"d9tsp9oddkk5lr s0cabg07u2m6";} timecreated: 1,550,833,015 origin: web ip: 133.8.69.30 realuserid: - join: { "name": "lo g", "parent": 604 } @timestamp: February 22nd 2019, 19:56:55.000 id: VOHbFGkBks2W33TXIzFE type: doc inde
February 22nd 2019, 19:56:55.000	id: 62,650 eventname: \core\event\course_viewed component: core action: viewed target: course objectt able: - objectid: - crud: r edulevel: 2 contextid: 2 contextlevel: 50 contextinstanceid: 1 user id: 0 courseid: 1 relateduserid: - anonymous: false other: N; timecreated: 1,550,833,015 origin: we b ip: 133.8.69.30 realuserid: - join: { "name": "log", "parent": 0 } @timestamp: February 22nd 2019, 19:56:55.00 0 id: VuHbFGkBks2W33TXIzFE type: doc index: moodle score: -
February 22nd 2019, 19:56:54.000	id: 62,644 eventname: \core\event\user_loggedout component: core action: loggedout target: user object table: user objectid: 602 crud: r edulevel: 0 contextid: 1 contextlevel: 10 contextinstanceid: 0 userid: 602 courseid: 0 relateduserid: - anonymous: false other: a:1:{s:9:"sessionid";s:26:"pnqs3ba1l01kq 2hc9ujtqtgudn";} timecreated: 1,550,833,014 origin: web ip: 133.8.69.30 realuserid: - join: { "name": "lo g", "parent": 602 } @timestamp: February 22nd 2019, 19:56:54.000 id: UOHbFGkBks2W33TXIzFE type: doc ind
February 22nd 2019, 19:56:54.000	id: 62,645 eventname: \core\event\user_loggedout component: core action: loggedout target: user object table: user objectid: 603 crud: r edulevel: 0 contextid: 1 contextlevel: 10 contextinstanceid: 0

Collapse

ユーザの特定


Literate Computing
for Reproducible Infrastructure
20181227_01_102-Elasticsearchの検索 Last Checkpoint: 2018/12/27 (autosaved)
Logout
File Edit View Insert Cell Kernel Navigate Widgets Help
Trusted
Python 3
Run
Markdown
lang:ALL
Contents
1 About: Elasticsearchの検索
2 準備
3 パスワードの取得
4 検索条件の入力
5 ログの検索
5.1 パラメータの指定
5.2 検索式を組み立てる
5.3 検索の実行
4 検索条件の入力
検索対象のキーバリューを入力してください。
[47]:
(例)
condition = {
"eventname.keyword": "\\tool_usertours\\event\\step_shown"
#}
condition = {
"eventname.keyword": "\\core\\event\\user_loggedin"
}
検索対象の基準となる日時を指定してください。なお、日時は日本時(JST)で入力されたものとみなします。
[48]:
# (例)
# target_time = '2018-12-07 13:00:00'
target_time ='2018-12-28 00:00:00'
基準となる日時からどの範囲までを検索対象とするかを指定します。指定方法の詳細についてはtimedeltaを参照してください。

5.3 検索の実行

検索を実行します。

[55]:
```
import requests
r = requests.get(
    es_url,
    json=es_search,
)
print(r)
```

```
<Response [200]>
```

検索条件に合致したログを表示します。

[56]:
```
%run scripts/data_masking.py

[
    (
        dm_decode(x['_source']['username'], password, salt),
        x['inner_hits']['log']['hits']['hits']
    )
    for x in r.json()['hits']['hits']
]
```

Out[56]:
```
[('admin',
  [{'_type': 'doc',
    '_id': 'OsiP8mcBks2W33TXypjJ',
    '_score': 0.0,
    '_routing': '2',
    '_source': {'id': 1021,
     'eventname': '\\core\\event\\user_loggedin',
     'component': 'core',
     'action': 'loggedin',
     'target': 'user',
     'objecttable': 'user',
     'objectid': 2,
     'crud': 'r',
     'edulevel': 0,
     'contextid': 1,
     'contextlevel': 10,
     'contextinstanceid': 0,
     'userid': 2,
     'courseid': 0,
     'relateduserid': None,
     'anonymous': False,
     'other': 'a:1:{s:8:"username";s:5:"admin";}',
     'timecreated': 1545960879,
     'origin': 'web',
     'ip': '133.8.69.30',
     'realuserid': None,
     'join': {'name': 'log', 'parent': 2},
     '@timestamp': '2018-12-28T01:34:39.000000000+00:00'}},
   {'_type': 'doc',
```

ユーザ名を復号して表示

admin

まとめ

- 学認クラウドオンデマンド構築サービスを利用して，オンプレミス（vmware）とパブリック（AWS）を融合したログ解析環境を構築した。
- クラウドに転送する個人情報は暗号化を行った。
- L2SW，DHCPログを統合管理するログ解析環境へと適用し，端末特定を行った。
- 監視データ（zabbix）とMoodleイベントログを統合管理するログ解析環境へと適用し，負荷テストの解析を行った。
  - イベントと負荷の関係が分かりやすく，障害対応や予兆検知への応用できる可能性が高い。

謝辞

- 本発表の研究内容は，平成30年度国立情報学研究所公募型共同研究の助成を受けています。