



Red Hat Enterprise Linux 9

IdM 상태 점검을 사용하여 IdM 환경 모니터링

IdM Healthcheck 유틸리티를 사용하여 ID 관리 서버 상태 모니터링

Red Hat Enterprise Linux 9 IdM 상태 점검을 사용하여 IdM 환경 모니터링

IdM Healthcheck 유틸리티를 사용하여 ID 관리 서버 상태 모니터링

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

법적 공지

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Using_IdM_Healthcheck_to_monitor_your_IdM_environment.ent file |.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution-Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

초록

이 문서 수집에서는 Red Hat Enterprise Linux 9에서 ID 관리를 효과적으로 구성, 관리 및 유지 관리하는 방법에 대한 지침을 제공합니다.

차례

보다 포괄적 수용을 위한 오픈 소스 용어 교체	3
RED HAT 문서에 관한 피드백 제공	4
1장. IDM 상태 점검 툴 설치 및 실행	5
1.1. IDM에서 상태 점검	5
1.2. IDM 상태 점검 설치	5
1.3. IDM 상태 점검 실행	6
1.4. 로그 교체	6
1.5. IDM 상태 점검을 사용하여 로그 회전 구성	6
1.6. 추가 리소스	7
2장. IDM 상태 점검을 사용하여 서비스 확인	9
2.1. 서비스 상태 점검 테스트	9
2.2. HEALTHCHECK를 사용하여 서비스 심사	9
3장. IDM 상태 점검을 사용하여 디스크 공간 확인	11
3.1. 디스크 공간 상태 점검 테스트	11
3.2. 상태 점검 툴을 사용하여 디스크 공간 표시	11
4장. HEALTHCHECK를 사용하여 IDM 구성 파일에 대한 권한 확인	13
4.1. 파일 권한 상태 점검 테스트	13
4.2. HEALTHCHECK를 사용하여 구성 파일 선별	14
5장. IDM 상태 점검을 사용하여 DNS 레코드 확인	16
5.1. DNS 레코드 상태 점검 테스트	16
5.2. 상태 확인 툴을 사용하여 DNS 레코드 표시	16
6장. 상태 점검을 사용하여 IDM 복제 확인	18
6.1. 복제 상태 테스트	18
6.2. HEALTHCHECK를 사용하여 복제 모니터링	19
7장. IDM 상태 점검을 사용하여 IDM 및 AD 신뢰 구성 확인	21
7.1. IDM 및 AD 신뢰 상태 점검 테스트	21
7.2. HEALTHCHECK 툴을 사용하여 신뢰 표시	22
8장. IDM 상태 점검을 사용하여 시스템 인증서 확인	24
8.1. 시스템 인증서 상태 점검 테스트	24
8.2. HEALTHCHECK를 사용하여 시스템 인증서 확인	25
9장. IDM 상태 점검을 사용하여 인증서 확인	27
9.1. IDM 인증서 상태 점검 테스트	27
9.2. HEALTHCHECK 도구를 사용하여 인증서 확인	29

보다 포괄적 수용을 위한 오픈 소스 용어 교체

Red Hat은 코드, 문서, 웹 속성에서 문제가 있는 용어를 교체하기 위해 최선을 다하고 있습니다. 먼저 마스터(master), 슬레이브(slave), 블랙리스트(blacklist), 화이트리스트(whitelist) 등 네 가지 용어를 교체하고 있습니다. 이러한 변경 작업은 작업 범위가 크므로 향후 여러 릴리스에 걸쳐 점차 구현할 예정입니다. 자세한 내용은 [CTO Chris Wright의 메시지](#)를 참조하십시오.

RED HAT 문서에 관한 피드백 제공

문서 개선을 위한 의견을 보내 주십시오. Red Hat이 어떻게 이를 개선할 수 있는지 알려 주십시오.

- 특정 문구에 대한 간단한 의견 작성 방법은 다음과 같습니다.
 1. 문서가 *Multi-page HTML* 형식으로 표시되는지 확인합니다. 또한 문서 오른쪽 상단에 **피드백** 버튼이 있는지 확인합니다.
 2. 마우스 커서를 사용하여 주석 처리하려는 텍스트 부분을 강조 표시합니다.
 3. 강조 표시된 텍스트 아래에 표시되는 **피드백 추가** 팝업을 클릭합니다.
 4. 표시된 지침을 따릅니다.
- Bugzilla를 통해 피드백을 제출하려면 새 티켓을 생성합니다.
 1. [Bugzilla](#) 웹 사이트로 이동합니다.
 2. 구성 요소로 **문서**를 사용합니다.
 3. **설명** 필드에 문서 개선을 위한 제안 사항을 기입하십시오. 관련된 문서의 해당 부분 링크를 알려주십시오.
 4. **버그 제출**을 클릭합니다.

1장. IDM 상태 점검 툴 설치 및 실행

이 장에서는 IdM 상태 점검 툴과 이를 설치 및 실행하는 방법에 대해 설명합니다.

1.1. IDM에서 상태 점검

IdM(Identity Management)의 Healthcheck 툴은 IdM 환경의 상태에 영향을 줄 수 있는 문제를 찾는 데 도움이 됩니다.



참고

Healthcheck 도구는 Kerberos 인증 없이 사용할 수 있는 명령줄 도구입니다.

모듈은 서로 독립적입니다.

Healthcheck는 테스트를 위한 독립적인 모듈로 구성되어 있습니다.

- 복제 문제
- 인증서 유효성
- 인증 기관 인프라 문제
- IdM 및 Active Directory 신뢰 문제
- 올바른 파일 권한 및 소유권 설정

두 개의 출력 형식

Healthcheck는 출력 유형 옵션을 사용하여 설정할 수 있는 다음 출력을 생성합니다.

- **JSON**: JSON 형식의 머신 읽을 수 있는 출력 (기본값)
- **사람**: 사람이 읽을 수 있는 출력

--output-file 옵션을 사용하여 다른 파일 대상을 지정할 수 있습니다.

결과

각 Healthcheck 모듈은 다음 결과 중 하나를 반환합니다.

SUCCESS

예상한 대로 구성

경고

실수는 아니지만 또는 평가할 가치가 있습니다.

ERROR

예상대로 구성되지 않았습니다.

심각

상당한 영향을 미칠 수 있으므로 예상대로 구성되지 않았습니다.

1.2. IDM 상태 점검 설치

이 섹션에서는 IdM 상태 점검 툴을 설치하는 방법을 설명합니다.

절차

- **ipa-healthcheck** 패키지를 설치합니다.

```
[root@server ~]# dnf install ipa-healthcheck
```

검증 단계

- **--failures-only** 옵션을 사용하여 **ipa-healthcheck** 만 보고 오류를 보고합니다. 완전히 작동하는 IdM 설치인 []의 빈 결과를 반환합니다.

```
[root@server ~]# ipa-healthcheck --failures-only
[]
```

추가 리소스

- **ipa-healthcheck --help** 를 사용하여 지원되는 모든 인수를 확인합니다.

1.3. IDM 상태 점검 실행

상태 점검은 [로그 회전을](#) 사용하여 수동으로 또는 자동으로 실행할 수 있습니다.

사전 요구 사항

- Healthcheck 툴이 설치되어 있어야 합니다. [IdM 상태 점검 설치](#)를 참조하십시오.

절차

- 수동으로 healthcheck을 실행하려면 **ipa-healthcheck** 명령을 입력합니다.

```
[root@server ~]# ipa-healthcheck
```

추가 리소스

모든 옵션에 대한 자세한 내용은 man page: **man ipa-healthcheck** 를 참조하십시오.

1.4. 로그 교체

로그 순환은 매일 새 로그 파일을 생성하고 파일은 날짜별로 구성됩니다. 로그 파일은 동일한 디렉터리에 저장되므로 날짜에 따라 특정 로그 파일을 선택할 수 있습니다.

교체란 최대 로그 파일의 수를 구성하고 번호가 초과되면 최신 파일을 다시 작성하고 가장 오래된 파일의 이름을 바꿉니다. 예를 들어 회전 번호가 30이면 첫 번째 로그 파일이 첫 번째 로그 파일인 첫 번째 로그 파일입니다.

로그 순환을 사용하면 비정상적인 로그 파일을 줄이고 이를 구성하므로 로그 분석에 도움이 될 수 있습니다.

1.5. IDM 상태 점검을 사용하여 로그 회전 구성

이 섹션에서는 다음을 사용하여 로그 교체를 구성하는 방법을 설명합니다.

- **systemd** 타이머
- **crond** 서비스

systemd 타이머는 Healthcheck 툴을 주기적으로 실행하고 로그를 생성합니다. 기본값은 매일 4 a.m로 설정됩니다.

crond 서비스는 로그 회전에 사용됩니다.

기본 로그 이름은 **healthcheck.log** 이고 순환된 로그는 **healthcheck.log-YYYYMMDD** 형식을 사용합니다.

사전 요구 사항

- root로 명령을 실행해야 합니다.

절차

1. **systemd** 타이머를 활성화합니다.

```
# systemctl enable ipa-healthcheck.timer
Created symlink /etc/systemd/system/multi-user.target.wants/ipa-healthcheck.timer ->
/usr/lib/systemd/system/ipa-healthcheck.timer.
```

2. **systemd** 타이머를 시작합니다.

```
# systemctl start ipa-healthcheck.timer
```

3. **/etc/logrotate.d/ipahealthcheck** 파일을 열어 저장해야 하는 로그 수를 구성합니다.
기본적으로 로그 회전은 30일 동안 설정됩니다.

4. **/etc/logrotate.d/ipahealthcheck** 파일에서 로그 경로를 구성합니다.
기본적으로 로그는 **/var/log/ipa/healthcheck/** 디렉터리에 저장됩니다.

5. **/etc/logrotate.d/ipahealthcheck** 파일에서 로그 생성 시간을 구성합니다.
기본적으로 로그는 매일 오전 4시 생성됩니다.

6. 로그 교체를 사용하려면 **crond** 서비스가 활성화되어 실행되고 있는지 확인합니다.

```
# systemctl enable crond
# systemctl start crond
```

로그 생성을 시작하려면 IPA 상태 점검 서비스를 시작합니다.

```
# systemctl start ipa-healthcheck
```

결과를 확인하려면 **/var/log/ipa/healthcheck/** 로 이동하여 로그가 올바르게 생성되었는지 확인합니다.

1.6. 추가 리소스

- IdM 상태 점검 사용 예를 보려면 [IdM 상태 점검 사용](#)의 다음 섹션을 참조하십시오.
 - [서비스 확인](#)

- IdM 및 AD 신뢰 구성 확인
- 인증서 확인
- 시스템 인증서 확인
- 디스크 공간 확인
- IdM 구성 파일에 대한 권한 확인
- 복제 확인

2장. IDM 상태 점검을 사용하여 서비스 확인

이 섹션에서는 Healthcheck 툴을 사용하여 IdM(Identity Management) 서버에서 사용하는 모니터링 서비스에 대해 설명합니다.

자세한 내용은 [IdM의 상태 점검](#)을 참조하십시오.

2.1. 서비스 상태 점검 테스트

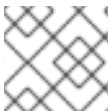
Healthcheck 툴에는 IdM 서비스가 실행되고 있지 않은지 확인하는 테스트가 포함되어 있습니다. 이 테스트는 실행되지 않는 서비스가 다른 테스트에서 실패할 수 있기 때문에 중요합니다. 따라서 모든 서비스가 먼저 실행 중인지 확인합니다. 그런 다음 다른 모든 테스트 결과를 확인할 수 있습니다.

모든 서비스 테스트를 보려면 **--list-sources** 옵션을 사용하여 **ipa-healthcheck** 을 실행합니다.

```
# ipa-healthcheck --list-sources
```

ipahealthcheck.meta.services 소스에서 Healthcheck로 테스트된 모든 서비스를 찾을 수 있습니다.

- certmonger
- dirsrv
- gssproxy
- httpd
- ipa_custodia
- ipa_dnssyncd
- ipa_otpd
- kadmin
- krb5kdc
- named
- pki_tomcatd
- sssd



참고

문제가 발견되면 모든 IdM 서버에서 이 테스트를 실행합니다.

2.2. HEALTHCHECK를 사용하여 서비스 심사

이 섹션에서는 Healthcheck 툴을 사용하여 IdM(Identity Management) 서버에서 실행되는 서비스에 대한 독립형 수동 테스트에 대해 설명합니다.

Healthcheck 툴에는 다음과 같이 결과가 단축될 수 있는 많은 테스트가 포함되어 있습니다.

- 모든 성공적인 테스트 제외: **--failures-only**

- 서비스 테스트만 포함: **--source=ipahealthcheck.meta.services**

절차

- 경고, 오류 및 서비스와 관련된 심각한 문제로 Healthcheck를 실행하려면 다음을 입력합니다.

```
# ipa-healthcheck --source=ipahealthcheck.meta.services --failures-only
```

테스트에 빈 대괄호가 표시됩니다.

```
[]
```

서비스 중 하나가 실패하면 결과는 다음 예와 유사합니다.

```
{
  "source": "ipahealthcheck.meta.services",
  "check": "httpd",
  "result": "ERROR",
  "kw": {
    "status": false,
    "msg": "httpd: not running"
  }
}
```

추가 리소스

- **man ipa-healthcheck** 를 참조하십시오.

3장. IDM 상태 점검을 사용하여 디스크 공간 확인

이 섹션에서는 Healthcheck 도구를 사용하여 Identity Management 서버의 사용 가능한 디스크 공간을 모니터링하는 방법에 대해 설명합니다.

자세한 내용은 [IDM의 상태 점검](#)을 참조하십시오.

3.1. 디스크 공간 상태 점검 테스트

Healthcheck 툴에는 사용 가능한 디스크 공간을 확인하는 테스트가 포함되어 있습니다. 디스크 여유 공간이 충분하지 않으면 다음과 같은 문제가 발생할 수 있습니다.

- 로깅
- 실행
- Backup

테스트에서는 다음 경로를 확인합니다.

표 3.1. 테스트된 경로

테스트에서 확인한 경로	최소 디스크 공간(MB)
<code>/var/lib/dirsrv/</code>	1024
<code>/var/lib/ipa/backup/</code>	512
<code>/var/log/</code>	1024
<code>var/log/audit/</code>	512
<code>/var/tmp/</code>	512
<code>/tmp/</code>	512

모든 테스트를 나열하려면 **--list-sources** 옵션을 사용하여 **ipa-healthcheck** 을 실행합니다.

```
# ipa-healthcheck --list-sources
```

파일 시스템 공간 검사 테스트는 **ipahealthcheck.system.filesystems-space** 소스 아래에 배치됩니다.

FileSystemSpaceCheck

이 테스트에서는 다음과 같은 방법으로 사용 가능한 디스크 공간을 확인합니다.

- 필요한 최소 원시 여유 바이트 수입니다.
- 최소 사용 가능한 디스크 공간은 20 %로 하드 코딩됩니다.

3.2. 상태 점검 툴을 사용하여 디스크 공간 표시

이 섹션에서는 Healthcheck 툴을 사용하여 IdM(Identity Management) 서버에서 사용 가능한 디스크 공간에 대한 독립 실행형 수동 테스트에 대해 설명합니다.

Healthcheck에는 많은 테스트가 포함되어 있으므로 다음과 같이 결과를 좁힐 수 있습니다.

- 모든 성공적인 테스트 제외: **--failures-only**
- 공간 검사 테스트만 포함: **--source=ipahealthcheck.system.filesystemspace**

절차

- 사용 가능한 디스크 공간과 관련된 경고, 오류 및 심각한 문제로 Healthcheck를 실행하려면 다음을 입력합니다.

```
# ipa-healthcheck --source=ipahealthcheck.system.filesystemspace --failures-only
```

테스트에 빈 대괄호가 표시됩니다.

```
[]
```

예를 들어 실패한 테스트가 표시될 수 있습니다.

```
{
  "source": "ipahealthcheck.system.filesystemspace",
  "check": "FileSystemSpaceCheck",
  "result": "ERROR",
  "kw": {
    "msg": "/var/lib/dirsrv: free space under threshold: 0 MiB < 1024 MiB",
    "store": "/var/lib/dirsrv",
    "free_space": 0,
    "threshold": 1024
  }
}
```

실패한 테스트는 **/var/lib/dirsrv** 디렉터리가 공간이 부족했음을 알려줍니다.

추가 리소스

- **man ipa-healthcheck** 를 참조하십시오.

4장. HEALTHCHECK를 사용하여 IDM 구성 파일에 대한 권한 확인

이 섹션에서는 Healthcheck 툴을 사용하여 IdM(Identity Management) 구성 파일을 테스트하는 방법을 설명합니다.

자세한 내용은 [IdM의 상태 점검](#)을 참조하십시오.

4.1. 파일 권한 상태 점검 테스트

Healthcheck 툴은 IdM(Identity Management)에서 설치 또는 구성한 일부 중요한 파일의 소유권 및 권한을 테스트합니다.

테스트된 파일의 소유권 또는 권한을 변경하면 테스트에서 **result** 섹션에 경고를 반환합니다. 구성이 작동하지 않지만 파일이 기본 구성과 다릅니다.

모든 테스트를 보려면 **--list-sources** 옵션을 사용하여 **ipa-healthcheck** 을 실행합니다.

```
# ipa-healthcheck --list-sources
```

파일 권한 테스트는 **ipahealthcheck.ipa.files** 소스 아래에 있습니다.

IPAFileNSSDBCheck

이 테스트에서는 389-ds NSS 데이터베이스 및 CA(인증 기관) 데이터베이스를 확인합니다. 389-ds 데이터베이스는 **/etc/dirsrv/slapd-*<dashed-REALM>***에 있으며 CA 데이터베이스는 **/etc/pki/pki-tomcat/alias/**에 있습니다.

IPAFileCheck

이 테스트에서는 다음 파일을 확인합니다.

- **/var/lib/ipa/ra-agent.{key|pem}**
- **/var/lib/ipa/certs/httpd.pem**
- **/var/lib/ipa/private/httpd.key**
- **/etc/httpd/alias/ipasession.key**
- **/etc/dirsrv/ds.keytab**
- **/etc/ipa/ca.crt**
- **/etc/ipa/custodia/server.keys**
PKINIT가 활성화된 경우:
- **/var/lib/ipa/certs/kdc.pem**
- **/var/lib/ipa/private/kdc.key**
DNS가 구성된 경우:
- **/etc/named.keytab**
- **/etc/ipa/dnssec/ipa-dnskeysyncd.keytab**

TomcatFileCheck

이 테스트에서는 CA가 구성된 경우 일부 tomcat 관련 파일을 확인합니다.

- `/etc/pki/pki-tomcat/password.conf`
- `/var/lib/pki/pki-tomcat/conf/ca/CS.cfg`
- `/etc/pki/pki-tomcat/server.xml`



참고

문제 찾기를 시도할 때 모든 IdM 서버에서 이러한 테스트를 실행합니다.

4.2. HEALTHCHECK를 사용하여 구성 파일 선별

이 섹션에서는 Healthcheck 툴을 사용하여 IdM(Identity Management) 서버의 구성 파일에 대한 독립형 수동 테스트를 설명합니다.

Healthcheck 툴에는 많은 테스트가 포함되어 있습니다. 결과는 다음과 같이 축소할 수 있습니다.

- 모든 성공적인 테스트 제외: **`--failures-only`**
- 소유권 및 권한 테스트만 포함: **`--source=ipahealthcheck.ipa.files`**

절차

1. IdM 구성 파일 소유권 및 권한에 대한 Healthcheck 테스트를 실행하려면 경고, 오류 및 심각한 문제만 표시하려면 다음을 입력합니다.

```
# ipa-healthcheck --source=ipahealthcheck.ipa.files --failures-only
```

테스트에 빈 대괄호가 표시됩니다.

```
# ipa-healthcheck --source=ipahealthcheck.ipa.files --failures-only
[]
```

실패한 테스트에서는 다음 경고와 유사한 결과를 표시합니다.

```
{
  "source": "ipahealthcheck.ipa.files",
  "check": "IPAFileNSSDBCheck",
  "result": "WARNING",
  "kw": {
    "key": "_etc_dirsrv_slapd-EXAMPLE-TEST_pkcs11.txt_mode",
    "path": "/etc/dirsrv/slapd-EXAMPLE-TEST/pkcs11.txt",
    "type": "mode",
    "expected": "0640",
    "got": "0666",
    "msg": "Permissions of /etc/dirsrv/slapd-EXAMPLE-TEST/pkcs11.txt are 0666 and should be 0640"
  }
}
```

추가 리소스

- **man ipa-healthcheck** 를 참조하십시오.

5장. IDM 상태 점검을 사용하여 DNS 레코드 확인

이 섹션에서는 **DNS** 레코드 문제를 식별하는 **IdM(Identity Management)**의 상태 점검 도구에 대해 설명합니다.

5.1. DNS 레코드 상태 점검 테스트

상태 점검 툴에는 자동 검색에 필요한 예상 **DNS** 레코드를 확인할 수 있는지 확인하는 테스트가 포함되어 있습니다.

모든 테스트를 나열하려면 **--list-sources** 옵션을 사용하여 **ipa-healthcheck** 을 실행합니다.

```
# ipa-healthcheck --list-sources
```

DNS 레코드 검사 테스트는 **ipahealthcheck.ipa.idns** 소스에 배치됩니다.

IPADNSSystemRecordsCheck

이 테스트에서는 **/etc/resolv.conf** 파일에 지정된 첫 번째 해결 방법을 사용하여 **ipa dns-update-system-records --dry-run** 명령에서 **DNS** 레코드를 확인합니다. 레코드는 **IPA** 서버에서 테스트합니다.

5.2. 상태 확인 툴을 사용하여 DNS 레코드 표시

이 섹션에서는 **Healthcheck** 툴을 사용하여 **IdM(Identity Management)** 서버의 **DNS** 레코드에 대한 독립형 수동 테스트를 설명합니다.

Healthcheck 툴에는 많은 테스트가 포함되어 있습니다. **--source ipahealthcheck.ipa.idns** 옵션을 추가하여 **DNS** 레코드 테스트만 포함하여 결과를 축소할 수 있습니다.

사전 요구 사항

- 상태 점검 테스트는 **root** 사용자로 수행해야 합니다.

절차

- DNS 레코드 검사를 실행하려면 다음을 입력합니다.

```
# ipa-healthcheck --source ipahealthcheck.ipa.idns
```

레코드를 확인할 수 있는 경우 테스트에서 **result**로 **SUCCESS** 를 반환합니다.

```
{
  "source": "ipahealthcheck.ipa.idns",
  "check": "IPADNSSystemRecordsCheck",
  "result": "SUCCESS",
  "uuid": "eb7a3b68-f6b2-4631-af01-798cac0eb018",
  "when": "20200415143339Z",
  "duration": "0.210471",
  "kw": {
    "key": "_ldap._tcp.idm.example.com.:server1.idm.example.com."
  }
}
```

예를 들어 레코드 수가 예상 수와 일치하지 않는 경우 테스트에서 **WARNING** 을 반환합니다.

```
{
  "source": "ipahealthcheck.ipa.idns",
  "check": "IPADNSSystemRecordsCheck",
  "result": "WARNING",
  "uuid": "972b7782-1616-48e0-bd5c-49a80c257895",
  "when": "20200409100614Z",
  "duration": "0.203049",
  "kw": {
    "msg": "Got {count} ipa-ca A records, expected {expected}",
    "count": 2,
    "expected": 1
  }
}
```

추가 리소스

- **man ipa-healthcheck** 를 참조하십시오.

6장. 상태 점검을 사용하여 IDM 복제 확인

이 섹션에서는 **Healthcheck** 툴을 사용하여 **IdM(Identity Management)** 복제를 테스트하는 방법을 설명합니다.

자세한 내용은 [IdM의 상태 점검](#)을 참조하십시오.

6.1. 복제 상태 테스트

Healthcheck 툴은 **IdM(Identity Management)** 토폴로지 구성을 테스트하고 복제 충돌 문제를 검색합니다.

모든 테스트를 나열하려면 **--list-sources** 옵션을 사용하여 **ipa-healthcheck** 을 실행합니다.

```
# ipa-healthcheck --list-sources
```

토폴로지 테스트는 **ipahealthcheck.ipa.topology** 및 **ipahealthcheck.ds.replication** 소스 아래에 배치됩니다.

IPATopologyDomainCheck

이 테스트에서는 다음을 검증합니다.

- 토폴로지의 연결이 끊어지지 않고 모든 서버 간에 복제 경로가 있습니다.
- 서버에 권장되는 복제 계약 수가 없는 경우.

테스트에 실패하면 테스트에서 연결 오류 또는 너무 많은 복제 계약과 같은 오류를 반환합니다.

테스트가 성공하면 테스트에서 구성된 도메인을 반환합니다.



참고

테스트는 도메인 및 **ca** 접미사 모두에 대해 **ipa topologysuffix-verify** 명령을 실행합니다(이 서버에 인증 기관을 구성하는 경우).

ReplicationConflictCheck

테스트에서는 **LDAP** 일치 (**&(!(!)=nsclass=nstombstone)(nsds5ReplConflict=*)**)의 항목을 검색합니다.



참고

문제를 확인하려고 할 때 모든 **IdM** 서버에서 이 테스트를 실행합니다.

6.2. HEALTHCHECK를 사용하여 복제 모니터링

이 섹션에서는 **Healthcheck** 도구를 사용하여 **IdM(Identity Management)** 복제 토폴로지 및 구성의 독립 실행형 수동 테스트를 설명합니다.

Healthcheck 툴에는 많은 테스트가 포함되어 있으므로 다음과 같이 결과를 단축할 수 있습니다.

- 복제 충돌 테스트: **--source=ipahealthcheck.ds.replication**
- 올바른 토폴로지 테스트: **--source=ipahealthcheck.ipa.topology**

사전 요구 사항

- 상태 점검 테스트는 **root** 사용자로 수행해야 합니다.

절차

- **Healthcheck** 복제 충돌 및 토폴로지 검사를 실행하려면 다음을 입력합니다.

```
# ipa-healthcheck --source=ipahealthcheck.ds.replication --
source=ipahealthcheck.ipa.topology
```

4 가지 다른 결과가 가능합니다:

- **SUCCESS-testd**가 성공적으로 통과되었습니다.

```
{
  "source": "ipahealthcheck.ipa.topology",
  "check": "IPATopologyDomainCheck",
  "result": "SUCCESS",
  "kw": {
    "suffix": "domain"
  }
}
```

- 경고: 테스트가 통과되었지만 문제가 있을 수 있습니다.

- **ERROR-databind-test**가 실패했습니다.

```
{
  "source": "ipahealthcheck.ipa.topology",
  "check": "IPATopologyDomainCheck",
  "result": "ERROR",
  "uuid": "d6ce3332-92da-423d-9818-e79f49ed321f",
  "when": "20191007115449Z",
  "duration": "0.005943",
  "kw": {
    "msg": "topologysuffix-verify domain failed, server2 is not connected
(server2_139664377356472 in MainThread)"
  }
}
```

- 테스트가 실패하고 **IdM** 서버 기능에 영향을 미칩니다.

추가 리소스

- **man ipa-healthcheck** 를 참조하십시오.

7장. IDM 상태 점검을 사용하여 IDM 및 AD 신뢰 구성 확인

이 섹션에서는 **IdM(Identity Management)**의 **Healthcheck** 툴을 이해하고 사용하여 **IdM** 및 **Active Directory** 신뢰 문제를 식별하는 데 도움이 됩니다.

7.1. IDM 및 AD 신뢰 상태 점검 테스트

Healthcheck 툴에는 **IdM(Identity Management)** 및 **AD(Active Directory)** 신뢰 상태를 테스트하는 데 필요한 여러 테스트가 포함되어 있습니다.

모든 신뢰 테스트를 확인하려면 **--list-sources** 옵션을 사용하여 **ipa-healthcheck** 을 실행합니다.

```
# ipa-healthcheck --list-sources
```

ipahealthcheck.ipa.trust 소스 아래에서 모든 테스트를 찾을 수 있습니다.

IPATrustAgentCheck

이 테스트에서는 시스템이 신뢰 에이전트로 구성된 경우 **SSSD** 설정을 확인합니다. **/etc/sss/sss.conf** 의 각 도메인에 대해 **id_provider=ipa** 가 **ipa_server_mode** 가 **True** 인지 확인합니다.

IPATrustDomainsCheck

이 테스트에서는 **sssctl domain-list** 의 도메인 목록을 **IPA** 도메인을 제외하고 **ipa trust-find** 의 도메인 목록과 비교하여 신뢰 도메인이 **SSSD** 도메인과 일치하는지 확인합니다.

IPATrustCatalogCheck

이 테스트에서는 **Administrator@REALM** 인 **AD** 사용자를 해결합니다. 그러면 **sssctl domain-status** 출력에 **AD Global** 카탈로그 및 **AD Domain Controller** 값이 채워집니다.

각 신뢰 도메인의 경우 **id**를 사용하여 **SID + 500(관리자)**의 사용자를 조회한 다음 **sssctl domain-status <domain> --active-server** 의 출력을 확인하여 도메인이 활성 상태인지 확인합니다.

IPAsidgenpluginCheck

이 테스트에서는 **IPA 389-ds** 인스턴스에서 **sidgen** 플러그인이 활성화되어 있는지 확인합니다. 또한 테스트에서는 **cn=plugins** 의 **IPA headGEN** 및 **ipa-sidgen-task** 플러그인에 **nsslapd-pluginEnabled** 옵션이 포함되어 있는지 확인합니다.

IPATrustAgentMemberCheck

이 테스트에서는 현재 호스트가 **cn=adtrust** 에이전트, **cn=sysaccounts**, **cn=etc**, **SUFFIX** 의 멤버인지 확인합니다.

IPATrustControllerPrincipalCheck

이 테스트에서는 현재 호스트가 **cn=adtrust** 에이전트, **cn=sysaccounts**, **cn=etc**, **SUFFIX** 의 멤버인지 확인합니다.

IPATrustControllerServiceCheck

이 테스트에서는 현재 호스트가 **ipactl**에서 **ADTRUST** 서비스를 시작하는지 확인합니다.

IPATrustControllerConfCheck

이 테스트에서는 **netconf** 목록의 출력에서 **passdb** 백엔드에 대해 **ldapi** 가 활성화되었는지 확인합니다.

IPATrustControllerGroupSIDCheck

이 테스트에서는 관리자 그룹의 **SID**가 **512(Domain Admins RID)**로 끝나는지 확인합니다.

IPATrustPackageCheck

이 테스트에서는 신뢰 컨트롤러 및 **AD** 트러스트가 활성화되지 않은 경우 **trust-ad** 패키지가 설치되어 있는지 확인합니다.



참고

문제를 해결하려고 할 때 모든 **IdM** 서버에서 이 테스트를 실행합니다.

7.2. HEALTHCHECK 툴을 사용하여 신뢰 표시

이 섹션에서는 **Healthcheck** 도구를 사용하여 **IdM(Identity Management)** 및 **AD(Active Directory)** 신뢰 상태 점검의 독립 실행형 수동 테스트를 설명합니다.

Healthcheck 툴에는 많은 테스트가 포함되어 있으므로 다음과 같이 결과를 단축할 수 있습니다.

- 모든 성공적인 테스트 제외: **--failures-only**
- 신뢰 테스트만 포함: **--source=ipahealthcheck.ipa.trust**

절차

- 경고, 오류 및 신뢰의 심각한 문제로 **Healthcheck**를 실행하려면 다음을 입력합니다.

```
# ipa-healthcheck --source=ipahealthcheck.ipa.trust --failures-only
```

테스트에 빈 대괄호가 표시됩니다.

```
# ipa-healthcheck --source=ipahealthcheck.ipa.trust --failures-only  
[]
```

추가 리소스

- **man ipa-healthcheck** 를 참조하십시오.

8장. IDM 상태 점검을 사용하여 시스템 인증서 확인

이 섹션에서는 **IdM(Identity Management)**의 상태 점검 툴을 설명하여 시스템 인증서 문제를 식별합니다.

자세한 내용은 [IdM의 상태 점검](#)을 참조하십시오.

8.1. 시스템 인증서 상태 점검 테스트

Healthcheck 툴에는 시스템(**DogTag**) 인증서를 확인하기 위한 여러 테스트가 포함되어 있습니다.

모든 테스트를 보려면 **--list-sources** 옵션을 사용하여 **ipa-healthcheck** 을 실행합니다.

```
# ipa-healthcheck --list-sources
```

ipahealthcheck.dogtag.ca 소스에서 모든 테스트를 찾을 수 있습니다.

DogtagCertsConfigCheck

이 테스트에서는 **NSS** 데이터베이스의 **CA(Certificate Authority)** 인증서를 **CS.cfg** 에 저장된 동일한 값과 비교합니다. 일치하지 않는 경우 **CA**가 시작되지 않습니다.

특히 확인을 수행합니다.

- **auditSigningCert cert-pki-ca against ca.audit_signing.cert**
- **ocspSigningCert cert-pki-ca against ca.ocsp_signing.cert**
- **caSigningCert cert-pki-ca against ca.signing.cert**
- **subsystemCert cert-pki-ca against ca.subsystem.cert**
- **server-Cert cert-pki-ca against ca.sslserver.cert**

키 복구 기관(KRA)이 설치된 경우:

- `transportCert cert-pki-kra against ca.connector.KRA.transportCert`

DogtagCertsConnectivityCheck

이 테스트에서는 연결을 확인합니다. 이 테스트는 검사를 수행하는 `ipa cert-show 1` 명령과 동일합니다.

- **Apache의 PKI 프록시 구성**
- **IdM에서 CA를 찾을 수 있음**
- **RA 에이전트 클라이언트 인증서**
- **요청에 대한 CA 응답의 수정**

`cert-show` 를 실행할 수 있는지 확인하고 **CA**(인증서 또는 찾을 수 없음)에서 예상 결과를 반환할 수 있으므로 테스트에서 직렬 #1로 인증서를 확인합니다.



참고

문제를 해결하려고 할 때 모든 IdM 서버에서 이 테스트를 실행합니다.

8.2. HEALTHCHECK를 사용하여 시스템 인증서 확인

이 섹션에서는 **Healthcheck** 툴을 사용하여 **IdM(Identity Management)** 인증서의 독립 실행형 수동 테스트에 대해 설명합니다.

상태 점검 툴에는 많은 테스트가 포함되어 있으므로 **DogTag tests: --source=ipahealthcheck.dogtag.ca**만 포함하면 결과를 좁힐 수 있습니다.

절차

- **Healthcheck restricted**를 **DogTag** 인증서로 제한하려면 다음을 입력합니다.

```
# ipa-healthcheck --source=ipahealthcheck.dogtag.ca
```

성공적인 테스트의 예:

```
{
  "source: ipahealthcheck.dogtag.ca",
  "check: DogtagCertsConfigCheck",
  "result: SUCCESS",
  "uuid: 9b366200-9ec8-4bd9-bb5e-9a280c803a9c",
  "when: 20191008135826Z",
  "duration: 0.252280",
  "kw:" {
    "key": "Server-Cert cert-pki-ca",
    "configfile": "/var/lib/pki/pki-tomcat/conf/ca/CS.cfg"
  }
}
```

실패한 테스트의 예는 다음과 같습니다.

```
{
  "source: ipahealthcheck.dogtag.ca",
  "check: DogtagCertsConfigCheck",
  "result: CRITICAL",
  "uuid: 59d66200-1447-4b3b-be01-89810c803a98",
  "when: 20191008135912Z",
  "duration: 0.002022",
  "kw:" {
    "exception": "NSDB /etc/pki/pki-tomcat/alias not initialized",
  }
}
```

추가 리소스

- **man ipa-healthcheck** 를 참조하십시오.

9장. IDM 상태 점검을 사용하여 인증서 확인

이 섹션에서는 **IdM(Identity Management)**에서 **Healthcheck** 툴을 이해하고 사용하여 **certmonger**에서 유지 관리하는 **IPA** 인증서의 문제를 식별하는 데 도움이 됩니다.

자세한 내용은 [IdM의 상태 점검](#)을 참조하십시오.

9.1. IDM 인증서 상태 점검 테스트

Healthcheck 툴에는 **IdM(Identity Management)**의 **certmonger**에서 유지 관리하는 인증서 상태를 확인하는 여러 테스트가 포함됩니다. **certmonger**에 대한 자세한 내용은 **certmonger**를 [사용하여 서비스에 대한 IdM 인증서 Obtaining an IdM certificate](#)을 참조하십시오.

이 테스트 검사 모음의 만료, 검증, 신뢰 및 기타 문제. 동일한 기본 문제에 대해 여러 오류가 발생할 수 있습니다.

모든 인증서 테스트를 보려면 **--list-sources** 옵션을 사용하여 **ipa-healthcheck**을 실행합니다.

```
# ipa-healthcheck --list-sources
```

ipahealthcheck.ipa.certs 소스에서 모든 테스트를 찾을 수 있습니다.

IPACertmongerExpirationCheck

이 테스트에서는 **certmonger**의 만료를 확인합니다.

오류가 보고되면 인증서가 만료되었습니다.

경고가 표시되면 인증서가 곧 만료됩니다. 기본적으로 이 테스트는 인증서가 만료되기 전 **28일** 이내에 적용됩니다.

/etc/ipahealthcheck/ipahealthcheck.conf 파일에서 일 수를 구성할 수 있습니다. 파일을 연 후 **default** 섹션에 있는 **cert_expiration_days** 옵션을 변경합니다.



참고

certmonger는 인증서 만료에 대한 자체 보기를 로드하고 유지합니다. 이 검사에서는 디스크상의 인증서의 유효성을 검사하지 않습니다.

IPACertfileExpirationCheck

이 테스트에서는 인증서 파일 또는 **NSS** 데이터베이스를 열 수 있는지 확인합니다. 이 테스트는 또한 만료를 확인합니다. 따라서 오류 또는 경고 출력에서 **msg** 속성을 주의 깊게 읽습니다. 메시지는 문제를 지정합니다.



참고

이 테스트에서는 디스크상의 인증서를 확인합니다. 인증서가 누락된 경우 읽을 수 없으므로 별도의 오류도 발생할 수 있습니다.

IPACertNSSTrust

이 테스트에서는 **NSS** 데이터베이스에 저장된 인증서에 대한 신뢰를 비교합니다. **NSS** 데이터베이스에서 예상되는 추적된 인증서의 경우 신뢰는 예상 값과 비교되며 일치하지 않는 항목에서 발생하는 오류입니다.

IPANSSChainValidation

이 테스트에서는 **NSS** 인증서의 인증서 체인의 유효성을 검사합니다. 테스트 실행: **certutil -V -u V -e -d [dbdir] -n [nickname]**

IPAOpenSSLChainValidation

이 테스트에서는 **OpenSSL** 인증서의 인증서 체인을 검증합니다. **NSSChain** 검증과 비교하려면 다음을 실행하는 **OpenSSL** 명령을 실행합니다.

```
openssl verify -verbose -show_chain -CAfile /etc/ipa/ca.crt [cert file]
```

IPARAAgent

이 테스트에서는 디스크의 인증서를 **uid=ipara,ou=People,o=ipaca**의 **LDAP**의 동등한 레코드와 비교합니다.

IPACertRevocation

이 테스트에서는 **certmonger**를 사용하여 인증서가 취소되지 않았는지 확인합니다. 따라서 이 테스트는 **certmonger**에서만 유지 관리하는 인증서와 관련된 문제를 찾을 수 있습니다.

IPACertmongerCA

이 테스트는 **certmonger CA**(인증 기관) 구성을 확인합니다. **IdM**은 **CA** 없이 인증서를 발행할 수 없습니다.

certmonger는 **CA** 도우미 세트를 유지합니다. **IdM**에는 **IdM**을 통해 인증서를 발행하고 호스트 또는 서비스 인증서에 대한 호스트 또는 사용자 주체로 인증하는 **IPA**라는 **CA**가 있습니다.

CA 하위 시스템 인증서를 갱신하는 **dogtag-ipa-ca-renew-agent** 및 **dogtag-ipa-ca-renew-agent-reuse** 도 있습니다.



참고

문제를 확인하려고 할 때 모든 **IdM** 서버에서 이 테스트를 실행합니다.

9.2. HEALTHCHECK 도구를 사용하여 인증서 확인

이 섹션에서는 **Healthcheck** 툴을 사용하여 **IdM**(Identity Management) 인증서 상태 점검의 독립 실행형 수동 테스트에 대해 설명합니다.

Healthcheck 툴에는 많은 테스트가 포함되어 있으므로 다음과 같이 결과를 단축할 수 있습니다.

- 모든 성공적인 테스트 제외: **--failures-only**
- 인증서 테스트만 포함: **--source=ipahealthcheck.ipa.certs**

사전 요구 사항

- 상태 점검 테스트는 **root** 사용자로 수행해야 합니다.

절차

- 인증서와 관련된 경고, 오류 및 심각한 문제로 **Healthcheck**를 실행하려면 다음을 입력합니다.

```
# ipa-healthcheck --source=ipahealthcheck.ipa.certs --failures-only
```

테스트에 빈 대괄호가 표시됩니다.

```
[]
```

실패한 테스트에서는 다음 출력을 보여줍니다.

```
{
  "source": "ipahealthcheck.ipa.certs",
  "check": "IPACertfileExpirationCheck",
  "result": "ERROR",
  "kw": {
    "key": 1234,
    "dbdir": "/path/to/nssdb",
    "error": [error],
    "msg": "Unable to open NSS database '/path/to/nssdb': [error]"
  }
}
```

이 IPACertfileExpirationCheck test가 NSS 데이터베이스를 여는 데 실패했습니다.

추가 리소스

- **man ipa-healthcheck** 를 참조하십시오.