

자체 호스팅 Linux 에이전트 구성하기

이 문서는 Azure DevOps 자체 호스팅 Linux 에이전트 구성 방법을 공유하기 위해 작성되었습니다.

※ 아래 내용은 Microsoft Docs에서 가져온 내용으로 출처는 하단 참조 링크에서 확인할 수 있습니다.

- 에이전트에 대해 알아보기
- 필수 구성 요소 확인
 - Subversion
 - TFVC
- 권한 준비
 - 자체 호스팅 에이전트에 대한 정보 보안
 - 사용할 사용자 결정
 - PAT(개인용 액세스 토큰)를 사용하여 인증
 - 사용자에게 사용 권한이 있는지 확인
- 에이전트 다운로드 및 구성
 - Azure Pipelines
 - 서버 URL
 - 인증 유형
 - Azure Pipelines
 - TFS 또는 Azure DevOps Server
- 대화형으로 실행
 - 한 번 실행
- 시스템 서비스로 실행
 - 명령
 - 에이전트 디렉터리로 변경
 - 설치
 - 시작
 - 상태
 - Stop
 - 제거
 - 환경 변수 업데이트
 - 서비스가 시작되기 전에 지침 실행
 - 서비스 파일
 - systemd 서비스 파일
 - .service 파일
 - 대체 서비스 메커니즘
- cgroup을 사용하여 에이전트 오류 방지
- 에이전트 교체
- 에이전트 제거 및 다시 구성
- 무인 구성
 - 필수 옵션
 - 인증 옵션
 - 풀 및 에이전트 이름
 - 에이전트 설정
 - Windows 전용 시작
 - 배포 그룹만
 - 환경만
- 진단
- 다른 옵션에 대한 도움말
- 기능
- FAQ
 - 최신 v2 에이전트 버전이 있는지 어떻게 할까요??
 - sudo가 서비스 명령을 실행하는 데 필요한 이유는 무엇인가요?
 - 방화벽을 실행하고 있으며 코드가 Azure Repos에 있습니다. 에이전트는 어떤 URL로 통신해야 하나요?
 - 자체 서명된 인증서를 사용하여 에이전트를 실행할 어떻게 할까요? 있나요?
 - 웹 프록시 뒤에서 에이전트를 실행할 어떻게 할까요? 있나요?
 - 에이전트를 다시 어떻게 할까요?
 - 웹 프록시를 우회하고 Azure Pipelines에 연결하도록 에이전트를 어떻게 할까요? 구성하시겠습니까?
 - 도메인을 *.visualstudio.com 사용하는 조직의 경우:
 - 도메인을 dev.azure.com 사용하는 조직의 경우:
 - TFS를 사용하고 있으며 위의 섹션에 있는 URL이 작동하지 않습니다. 어디에서 도움을 받을 수 있나요?
- 참고

작업을 실행하려면 하나 이상의 에이전트가 필요합니다. Linux 에이전트는 Java 및 Android 앱을 포함하여 다양한 종류의 앱을 빌드하고 배포할 수 있습니다. Ubuntu, Red Hat 및 CentOS를 지원합니다.

시작하기 전에

- **파이프라인이 Azure Pipelines에 있고 Microsoft 호스팅 에이전트가 요구 사항을 충족하는 경우 프라이빗 Linux 에이전트 설정을 건너뛸 수 있습니다.**
- 그렇지 않으면 Linux에서 에이전트를 설정할 수 있는 올바른 위치로 이동했습니다. 다음 섹션을 계속합니다.

에이전트에 대해 알아보기

에이전트가 무엇이고 어떻게 작동하는지 이미 알고 있는 경우 다음 섹션으로 바로 이동할 수 있습니다. 그러나 수행하는 작업 및 작동 방식에 대한 자세한 배경 지식은 [Azure Pipelines 에이전트를 참조하세요](#).

필수 구성 요소 확인

에이전트는 .NET Core 3.1을 기반으로 합니다. 이 에이전트는 여러 Linux 배포판에서 실행할 수 있습니다. 지원되는 .NET Core 배포판의 하위 집합은 다음과 같습니다.

- X64
 - CentOS 7, 6(참고 1 참조)
 - Debian 9
 - 페도라 30, 29
 - Linux Mint 18, 17
 - openSUSE 42.3 이상
 - Oracle Linux 7
 - Red Hat Enterprise Linux 8, 7, 6(참고 1 참조)
 - SUSE Enterprise Linux 12 SP2 이상
 - Ubuntu 20.04, 18.04, 16.04
 - CBL-Mariner 1.0(참고 3 참조)
- ARM32(참고 2 참조)
 - Debian 9
 - Ubuntu 18.04
- ARM64
 - Debian 9
 - Ubuntu 21.04, 20.04, 18.04

참고

참고 1: RHEL 6 및 CentOS 6에는 특수 `rhel.6-x64` 버전의 에이전트를 설치해야 합니다.

참고

참고 2: ARM 명령 집합 `ARMv7` 이상이 필요합니다. 실행 `uname -a` 하여 Linux 배포판의 명령 집합을 확인합니다.

참고

Mariner OS 배포는 현재 Azure DevOps 에이전트에서 부분적으로 지원됩니다. 스크립트에서 `installdependencies.sh` 이 OS 배포를 검색하기 위한 메커니즘을 제공하고 있지만 .NET Core 쪽의 지원이 부족하여 이 OS 배포판에서 실행할 때 모든 에이전트 함수의 완전한 작동성을 보장할 수 없습니다.

플랫폼에 관계없이 Git 2.9.0 이상을 설치해야 합니다. 최신 버전의 Git을 설치하는 것이 좋습니다.

참고

에이전트 설치 관리자는 다른 종속성을 확인하는 방법을 알고 있습니다. 에이전트 디렉터리에서 실행 `./bin/installdependencies.sh` 하여 지원되는 Linux 플랫폼에 이러한 종속성을 설치할 수 있습니다.

.NET Core에 필요한 이러한 종속성 중 일부는 다음과 같은 packages.fficios.com 타사 사이트에서 가져옵니다. 스크립트를 검토하고 스크립트를 `installdependencies.sh` 실행하기 전에 Linux 컴퓨터에서 참조된 타사 사이트에 액세스할 수 있는지 확인합니다.

또한 필요한 모든 리포지토리가 (예: `apt` 또는 `zypper`) 사용되는 관련 패키지 관리자에 `installdependencies.sh` 연결되어 있는지 확인하세요.

종속성 설치 문제(예: '리포지토리에서 종속성을 찾을 수 없음' 또는 '리포지토리 인덱스 파일 검색 문제')의 경우 배포 소유자에게 연락하여 추가 지원을 요청하면 됩니다.

Subversion

Subversion 리포지토리에서 빌드하는 경우 컴퓨터에 Subversion 클라이언트를 설치해야 합니다.

에이전트 설치 프로그램을 처음 수동으로 실행해야 합니다. 에이전트 작동 방식에 대한 느낌을 받거나 많은 에이전트 설정을 자동화하려는 경우 [무인 구성](#)을 사용하는 것이 좋습니다.

TFVC

TFVC를 사용하는 경우 [Oracle Java JDK 1.6](#) 이상이 필요합니다. (Oracle JRE 및 OpenJDK는 이 용도로 충분하지 않습니다.)

TEE 플러그 인 은 TFVC 기능에 사용됩니다. 구성 중에 TFVC를 사용하려는 경우 수락해야 하는 EULA가 있습니다.

TEE 플러그 인은 더 이상 유지 관리되지 않으며 오래된 Java 종속성을 포함하므로 에이전트 2.198.0부터는 더 이상 에이전트 배포에 포함되지 않습니다. 그러나 TFVC 리포지토리를 체크 아웃하는 경우 체크 아웃 작업 실행 중에 TEE 플러그 인이 다운로드됩니다. TEE 플러그 인은 작업 실행 후에 제거됩니다.

참고

참고: 이 다운로드 메커니즘으로 인해 체크 아웃 작업이 작업을 시작하는 데 시간이 오래 걸리는 것을 알 수 있습니다.

에이전트가 프록시 또는 방화벽 뒤에서 실행되는 경우 다음 사이트에 <https://vstsagenttools.blob.core.windows.net/> 대한 액세스를 확인해야 합니다. TEE 플러그 인은 이 주소에서 다운로드됩니다.

자체 호스팅 에이전트를 사용하고 TEE 다운로드와 관련된 문제가 발생하는 경우 TEE를 수동으로 설치할 수 있습니다.

1. 환경 또는 파이프라인 변수를 `.true` 설정합니다 `DISABLE_TEE_PLUGIN_REMOVAL`. 이 변수는 TFVC 리포지토리 체크 아웃 후 에이전트가 TEE 플러그 인을 제거하는 것을 방지합니다.
2. [Team Explorer Everywhere GitHub 릴리스](#)에서 TEE-CLC 버전 14.135.0을 수동으로 다운로드합니다.
3. 폴더의 내용을 추출합니다 `TEE-CLC-14.135.0<agent_directory>/externals/tee`.

권한 준비

자체 호스팅 에이전트에 대한 정보 보안

에이전트를 구성하는 사용자에게는 풀 관리자 권한이 필요하지만 에이전트를 실행하는 사용자는 그렇지 않습니다.

에이전트에서 제어하는 폴더는 가능한 한 적은 수의 사용자로 제한되어야 하며 암호 해독 또는 유출될 수 있는 비밀을 포함해야 합니다.

ADO 파이프라인 에이전트는 외부 소스에서 다운로드하는 코드를 실행하도록 설계된 소프트웨어 제품입니다. 기본적으로 RCE(원격 코드 실행) 공격의 대상이 될 수 있습니다.

따라서 파이프라인 에이전트의 각 개별 사용을 둘러싼 위협 모델을 고려하여 작업을 수행하고 에이전트를 실행하는 사용자, 에이전트가 실행되는 컴퓨터, 파이프라인 정의에 대한 쓰기 액세스 권한이 있는 사용자, yaml이 저장된 git 리포지토리에 부여할 수 있는 최소 권한을 결정하는 것이 중요합니다. 또는 새 파이프라인에 대한 풀에 대한 액세스를 제어하는 사용자 그룹입니다.

에이전트를 실행하는 ID가 에이전트를 풀에 연결할 수 있는 권한이 있는 ID와 달라지도록 하는 것이 가장 좋습니다.

자격 증명(및 기타 에이전트 관련 파일)을 생성하는 사용자는 자격 증명을 읽어야 하는 사용자와 다릅니다.

따라서 에이전트 컴퓨터 자체에 부여된 액세스 권한과 로그 및 아티팩트와 같은 중요한 파일이 포함된 에이전트 폴더를 신중하게 고려하는 것이 더 안전합니다.

DevOps 관리자 및 에이전트 프로세스를 실행하는 사용자 ID에 대해서만 에이전트 폴더에 대한 액세스 권한을 부여하는 것이 좋습니다.

관리자는 빌드 실패를 이해하거나 Azure DevOps 오류를 보고할 수 있도록 로그 파일을 가져오기 위해 파일 시스템을 조사해야 할 수 있습니다.

사용할 사용자 결정

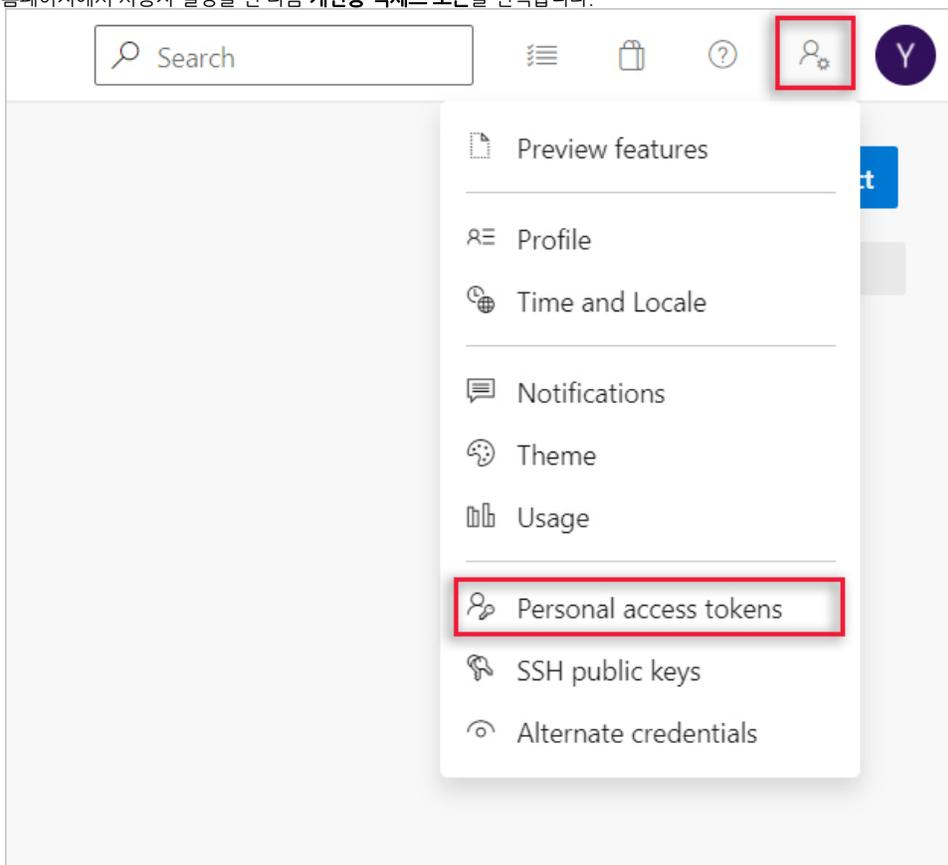
일회성 단계로 에이전트를 등록해야 합니다. [에이전트 큐를 관리](#)할 수 있는 권한이 있는 사용자는 다음 단계를 완료해야 합니다.

에이전트는 일상적인 작업에서 이 사용자의 자격 증명을 사용하지 않지만 등록을 완료해야 합니다. [에이전트가 통신하는 방법에](#) 대해 자세히 알아봅니다.

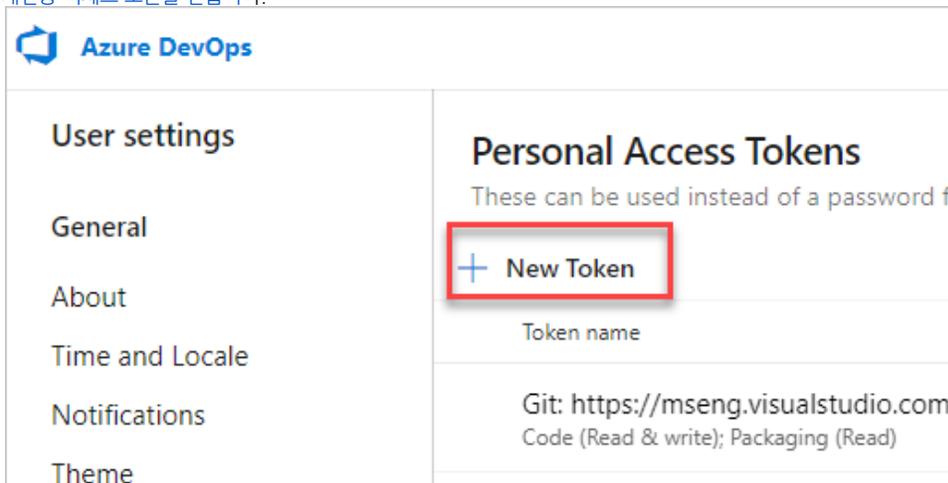
PAT(개인용 액세스 토큰)를 사용하여 인증

1. Azure DevOps 조직(https://dev.azure.com/{your_organization})에서 사용할 사용자 계정으로 로그인합니다.

- 홈페이지에서 사용자 설정을 연 다음 **개인용 액세스 토큰**을 선택합니다.



- 개인용 액세스 토큰을 만듭니다.



- 범위에 대해 **에이전트 풀(읽기, 관리)**을 선택하고 다른 모든 상자가 선택 취소되었는지 확인합니다.
배포 그룹 에이전트인 경우 범위에 대해 **배포 그룹(읽기, 관리)**을 선택하고 다른 모든 상자가 지워지는지 확인합니다.
새 개인용 액세스 토큰 만들기 창의 맨 아래에 있는 **모든 범위 표시**를 선택하여 전체 범위 목록을 확인합니다.
- 토큰을 복사합니다. 에이전트를 구성할 때 이 토큰을 사용합니다.

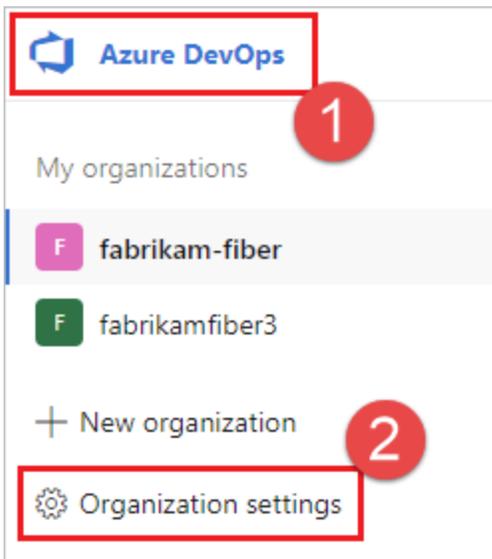
사용자에게 사용 권한이 있는지 확인

사용하려는 사용자 계정에 에이전트를 등록할 수 있는 권한이 있는지 확인합니다.

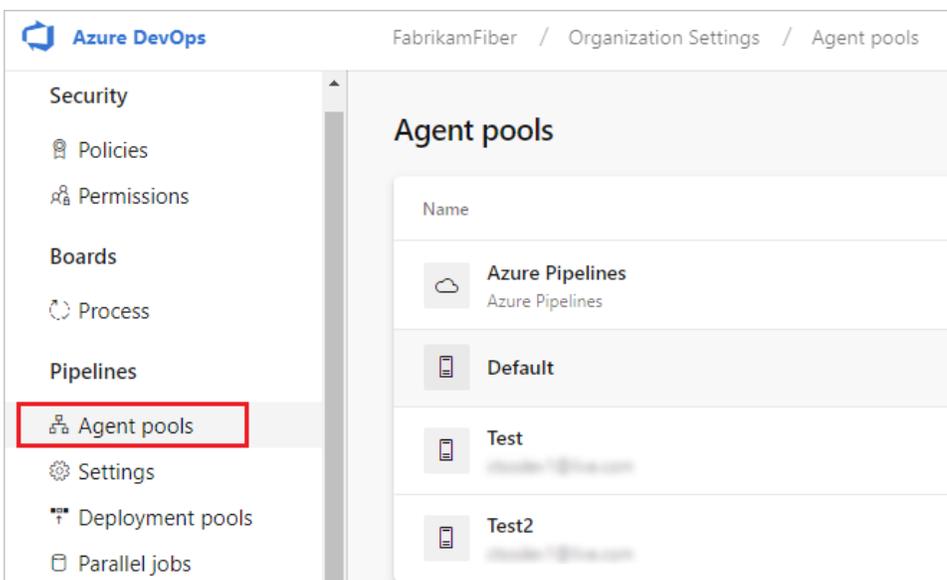
사용자가 Azure DevOps 조직 소유자 또는 TFS 또는 Azure DevOps Server 관리자인가요? **여기서 중지**합니다. 권한이 있습니다.

그렇지 않은 경우:

- 브라우저를 열고 Azure Pipelines 조직 또는 Azure DevOps Server 또는 TFS 서버의 **에이전트 풀** 탭으로 이동합니다.
 - Azure DevOps, 조직 설정**을 선택합니다.



b. 에이전트 풀을 선택합니다.



- 페이지 오른쪽에 있는 풀을 선택한 다음 **보안**을 클릭합니다.
- 사용하려는 사용자 계정이 표시되지 않으면 관리자가 계정을 추가하도록 합니다.
관리자는 에이전트 풀 관리자, [Azure DevOps 조직 소유자](#) 또는 [TFS 또는 Azure DevOps Server 관리자](#)일 수 있습니다.

배포 그룹 에이전트인 경우 관리자는 [배포 그룹 관리자](#), [Azure DevOps 조직 소유자](#) 또는 [TFS 또는 Azure DevOps Server 관리자](#)가 될 수 있습니다.

[Azure Pipelines](#)의 배포 그룹 페이지에 있는 **보안** 탭에서 **배포 그룹 관리자** 역할에 사용자를 추가할 수 있습니다.

i 참고

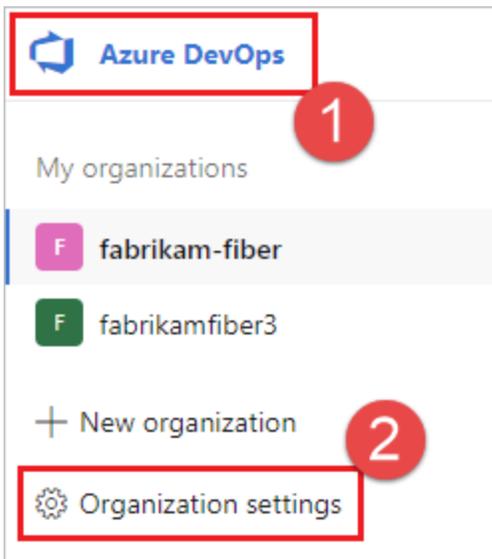
다음과 같은 메시지가 표시되면 **ID를 추가할 수 없습니다. 다른 ID를 사용해 보세요.**
조직 소유자 또는 TFS 또는 Azure DevOps Server 관리자를 위해 위의 단계를 수행했을 것입니다.
아무 작업도 수행할 필요가 없습니다. 에이전트 큐를 관리할 수 있는 권한이 이미 있습니다.

에이전트 다운로드 및 구성

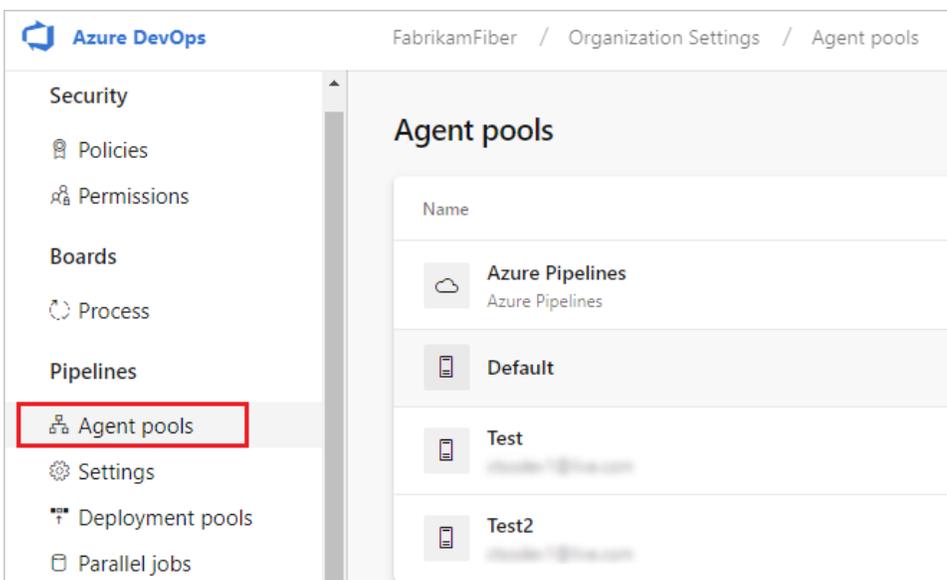
Azure Pipelines

- 위에서 설명한 대로 사용 권한을 준비한 계정을 사용하여 컴퓨터에 로그인합니다.
- 웹 브라우저에서 Azure Pipelines에 로그인하고 **에이전트 풀** 탭으로 이동합니다.

a. **Azure DevOps, 조직 설정**을 선택합니다.



b. 에이전트 풀을 선택합니다.



3. 기본 풀을 선택하고 에이전트 탭을 선택한 다음 새 에이전트를 선택합니다.
4. 에이전트 가져오기 대화 상자에서 Linux를 클릭합니다.
5. 왼쪽 창에서 특정 맛을 선택합니다. 대부분의 Linux 배포판에 x64 또는 ARM을 제공합니다. 또한 Red Hat Enterprise Linux 6에 대한 특정 빌드를 제공합니다.
6. 오른쪽 창에서 다운로드 단추를 클릭합니다.
7. 페이지의 지침을 따릅니다.
8. 선택한 디렉터리에 에이전트의 압축을 풉니다. cd 해당 디렉터리로 이동하여 실행 ./config.sh합니다.

서버 URL

Azure Pipelines: <https://dev.azure.com/{your-organization}>

인증 유형

Azure Pipelines

PAT를 선택한 다음, 만든 PAT 토큰을 명령 프롬프트 창에 붙여넣습니다.



참고

PAT를 인증 방법으로 사용하는 경우 PAT 토큰은 에이전트의 초기 구성에만 사용됩니다. [Azure Pipelines 또는 TFS와의 통신](#)에 대해 자세히 알아보세요.

TFS 또는 Azure DevOps Server

i **중요**

사용하려는 인증 방법을 지원하도록 서버가 구성되어 있는지 확인합니다.

TFS에 연결하도록 에이전트를 구성하는 경우 다음 옵션이 제공됩니다.

- **대체** 기본 인증을 사용하여 TFS 또는 Azure DevOps Server 연결합니다. 대체를 선택하면 자격 증명을 묻는 메시지가 표시됩니다.
- **통합** macOS 또는 Linux에서는 지원되지 않습니다.
- **협상**(기본값) NTLM 또는 Kerberos와 같은 Windows 인증 체계를 통해 로그인한 사용자 이외의 사용자로 TFS 또는 Azure DevOps Server 연결합니다. 협상을 선택하면 자격 증명을 묻는 메시지가 표시됩니다.
- **패트** Azure Pipelines 및 TFS 2017 이상에서만 지원됩니다. PAT를 선택한 후 **만든 PAT 토큰**을 명령 프롬프트 창에 붙여넣습니다. Azure DevOps Server 또는 TFS 인스턴스와 에이전트 컴퓨터가 신뢰할 수 있는 도메인에 없는 경우 PAT(개인용 액세스 토큰)를 사용합니다. PAT 인증은 도메인 컨트롤러 대신 Azure DevOps Server 또는 TFS 인스턴스에서 처리됩니다.

i **참고**

PAT를 인증 방법으로 사용하는 경우 PAT 토큰은 Azure DevOps Server 에이전트의 초기 구성 및 최신 버전의 TFS에만 사용됩니다. [Azure Pipelines 또는 TFS와의 통신](#)에 대해 자세히 알아보세요.

대화형으로 실행

대화형 모드 또는 서비스로 에이전트를 실행할지 여부에 대한 지침은 [에이전트: 대화형 대 서비스를](#) 참조하세요.

에이전트를 대화형으로 실행하려면 다음을 수행합니다.

1. 에이전트를 서비스로 실행한 경우 [서비스를 제거](#)합니다.
2. 에이전트를 실행합니다.

```
./run.sh
```

에이전트를 다시 시작하려면 Ctrl+C를 누른 다음 실행 run.sh 하여 다시 시작합니다.

에이전트를 사용하려면 에이전트의 풀을 사용하여 [작업](#)을 실행합니다. 다른 풀을 선택하지 않은 경우 에이전트는 **기본 풀**에 있습니다.

한 번 실행

대화형으로 실행하도록 구성된 에이전트의 경우 에이전트가 하나의 작업만 허용하도록 선택할 수 있습니다. 이 구성에서 실행하려면 다음을 수행합니다.

```
./run.sh --once
```

이 모드의 에이전트는 하나의 작업만 수락한 다음 정상적으로 스킨다운합니다(Azure Container Instances 같은 서비스에서 [Docker](#)에서 실행하는 데 유용).

시스템 서비스로 실행

에이전트가 이러한 운영 체제에서 실행되는 경우 에이전트를 서비스로 systemd 실행할 수 있습니다.

- Ubuntu 16 LTS 이상
- Red Hat 7.1 이상

에이전트를 서비스로 실행하고 관리하기 위한 예제 `./svc.sh` 스크립트를 systemd 제공합니다.

이 스크립트는 에이전트를 구성한 후에 생성됩니다. 스크립트를 실행하기 전에 검토하고 필요한 경우 업데이트하는 것이 좋습니다.

몇 가지 중요한 주의 사항:

- 에이전트를 서비스로 실행하는 경우 에이전트 서비스를 사용자로 root 실행할 수 없습니다.
- [SELinux](#)를 실행하는 사용자는 제공된 스크립트에 어려움을 보고했습니다.svc.sh. [이 에이전트 문제를](#) 시작점으로 참조하세요. SELinux는 공식적으로 지원되는 구성이 아닙니다.

i **참고**

다른 배포가 있거나 다른 방법을 선호하는 경우 원하는 서비스 메커니즘을 사용할 수 있습니다. [서비스 파일을](#) 참조하세요.

명령

에이전트 디렉터리로 변경

예를 들어 홈 디렉터리의 하위 폴더에 myagent 설치한 경우:

```
cd ~/myagent$
```

설치

명령:

```
sudo ./svc.sh install [username]
```

이 명령은 을 가리키는 ./runsvc.sh 서비스 파일을 만듭니다. 이 스크립트는 환경(아래 세부 정보)을 설정하고 에이전트 호스트를 시작합니다. 매개 변수를 지정하지 않으면 username sudo 명령에 의해 설정된 \$SUDO_USER 환경 변수에서 사용자 이름을 가져옵니다. 이 변수는 항상 명령을 호출 sudo 한 사용자의 이름과 같습니다.

시작

```
sudo ./svc.sh start
```

상태

```
sudo ./svc.sh status
```

Stop

```
sudo ./svc.sh stop
```

제거

제거하기 전에 중지해야 합니다.

```
sudo ./svc.sh uninstall
```

환경 변수 업데이트

서비스를 구성할 때 PATH, LANG, JAVA_HOME, ANT_HOME 및 MYSQL_PATH 같은 현재 로그인 사용자에게 대한 몇 가지 유용한 환경 변수의 스냅샷을 만듭니다. 변수를 업데이트해야 하는 경우(예: 새 소프트웨어를 설치한 후)

```
./env.sh
sudo ./svc.sh stop
sudo ./svc.sh start
```

환경 변수의 스냅샷은 에이전트 루트 디렉터리 아래에 있는 파일에 PATH 저장 .env 되며 .path, 이러한 파일을 직접 변경하여 환경 변수 변경 내용을 적용할 수도 있습니다.

서비스가 시작되기 전에 지침 실행

서비스가 시작될 때 실행하기 위해 사용자 고유의 지침과 명령을 실행할 수도 있습니다. 예를 들어 환경을 설정하거나 스크립트를 호출할 수 있습니다.

1. runsvc.sh을 편집합니다.
2. 다음 줄을 지침으로 바꿉니다.

```
# insert anything to setup env when running as a service
```

서비스 파일

서비스를 설치하면 일부 서비스 파일이 배치됩니다.

systemd 서비스 파일

시스템 서비스 파일이 만들어집니다.

```
/etc/systemd/system/vsts.agent.{tfs-name}.{agent-name}.service
```

예를 들어 이름을 our-linux-agent 사용하여 에이전트(위 참조)를 구성했습니다. 서비스 파일은 다음 중 하나가 됩니다.

- **Azure Pipelines:** 조직의 이름입니다. 예를 들어 연결하는 <https://dev.azure.com/fabrikam> 경우 서비스 이름은 입니다.
/etc/systemd/system/vsts.agent.fabrikam.our-linux-agent.service
- **TFS 또는 Azure DevOps Server:** 온-프레미스 서버의 이름입니다. 예를 들어 연결하는 <http://our-server:8080/tfs> 경우 서비스 이름은 입니다.
/etc/systemd/system/vsts.agent.our-server.our-linux-agent.service

```
sudo ./svc.sh install 이 템플릿에서 이 파일을 생성합니다. ./bin/vsts.agent.service.template
```

.service 파일

```
sudo ./svc.sh start 는 위에서 설명한 시스템 서비스 파일의 이름을 포함하는 파일을 읽 .service 어 서비스를 찾습니다.
```

대체 서비스 메커니즘

에이전트를 ./svc.sh 시스템 서비스로 실행하고 관리할 수 있는 편리한 방법으로 스크립트를 제공합니다. 하지만 원하는 서비스 메커니즘(예: initd 또는 upstart)을 사용할 수 있습니다.

위에서 설명한 템플릿을 사용하여 다른 종류의 서비스 파일을 쉽게 생성할 수 있습니다.

cgroup을 사용하여 에이전트 오류 방지

그렇지 않으면 에이전트가 파이프라인 로그를 스트리밍하거나 파이프라인 상태를 서버에 다시 보고할 수 없으므로 에이전트가 실패하거나 사용할 수 없게 되는 상황을 방지하는 것이 중요합니다.

cgroups 및 더 낮은 oom_score_adj 메모리 압력을 사용하여 이러한 종류의 문제가 발생할 위험을 완화할 수 있습니다.

이 작업을 수행한 후 Linux는 에이전트 프로세스에서 메모리를 회수하기 전에 파이프라인 작업 프로세스에서 시스템 메모리를 회수합니다. [cgroups 및 OOM 점수를 구성하는 방법을 알아봅니다.](#)

에이전트 교체

에이전트를 바꾸려면 다운로드를 수행하고 에이전트 단계를 다시 구성합니다.

이미 존재하는 에이전트와 동일한 이름을 사용하여 에이전트를 구성할 때 기존 에이전트를 바꿀 것인지 묻는 메시지가 표시됩니다. 대답 y하는 경우 대체하려는 에이전트(아래 참조)를 제거해야 합니다. 그렇지 않으면 몇 분 동안 충돌이 발생하면 에이전트 중 하나가 종료됩니다.

에이전트 제거 및 다시 구성

에이전트를 제거하려면

1. 위에서 설명한 대로 서비스를 중지하고 제거합니다.
2. 에이전트를 제거합니다.

```
./config.sh remove
```

3. 자격 증명을 입력합니다.

에이전트를 제거한 후 **다시 구성**할 수 있습니다.

무인 구성

사람이 개입하지 않고 스크립트에서 에이전트를 설정할 수 있습니다. 모든 질문에 대한 답변과 대답을 전달 `--unattended` 해야 합니다.

에이전트를 구성하려면 조직의 URL 또는 에이전트를 설정할 권한이 있는 사람의 컬렉션 및 자격 증명을 알고 있어야 합니다. 다른 모든 응답은 선택 사항입니다. 대신 환경 변수를 사용하여 명령줄 매개 변수를 지정할 수 있습니다. 대문자로 이름을 입력하고 앞에 추가합니다 `VSTS_AGENT_INPUT_`. 예를 들어 `VSTS_AGENT_INPUT_PASSWORD--password`

필수 옵션

- `--unattended` - 에이전트 설정에서 정보를 입력하라는 메시지가 표시되지 않으며 명령줄에서 모든 설정을 제공해야 합니다.
- `--url <url>` - 서버의 URL입니다. 예: <https://dev.azure.com/myorganization> 또는 <http://my-azure-devops-server:8080/tfs>
- `--auth <type>` - 인증 유형입니다. 유효한 값은 다음과 같습니다.
 - `pat` (개인용 액세스 토큰) - PAT는 Azure DevOps Services 작동하는 유일한 체계입니다.
 - `negotiate` (Kerberos 또는 NTLM)
 - `alt` (기본 인증)
 - `integrated` (Windows 기본 자격 증명)

인증 옵션

- 선택한 `--auth pat` 경우:
 - `--token <token>` - 개인 액세스 토큰 지정
 - PAT는 Azure DevOps Services 작동하는 유일한 체계입니다.
- 선택한 경우 또는 `--auth alt` 다음을 수행 `--auth negotiate` 합니다.
 - `--userName <userName>` - 형식 `domain\userName` 으로 Windows 사용자 이름을 지정하거나 `userName@domain.com`
 - `--password <password>` - 암호를 지정합니다.

풀 및 에이전트 이름

- `--pool <pool>` - 조인할 에이전트의 풀 이름
- `--agent <agent>` - 에이전트 이름
- `--replace` - 풀의 에이전트를 대체합니다. 다른 에이전트가 동일한 이름으로 수신 대기하는 경우 충돌로 인해 실패하기 시작합니다.

에이전트 설정

- `--work <workDirectory>` - 작업 데이터가 저장되는 작업 디렉터리입니다. `_work` 기본적으로 에이전트 디렉터리의 루트 아래에 있습니다. 작업 디렉터리가 지정된 에이전트에 의해 소유되며 여러 에이전트 간에 공유해서는 안 됩니다.
- `--acceptTeeEula` - Team Explorer Everywhere 최종 사용자 사용권 계약 수락 (macOS 및 Linux에만 해당)
- `--disableloguploads` - 서버로 콘솔 로그 출력을 스트리밍하거나 보내지 마세요. 대신 작업이 완료된 후 에이전트 호스트의 파일 시스템으로부터 검색할 수 있습니다.

Windows 전용 시작

- `--runAsService` - Windows 서비스로 실행되도록 에이전트 구성 (관리자 권한 필요)
- `--runAsAutoLogon` - 자동 로그인 구성 및 시작 시 에이전트 실행 (관리자 권한 필요)
- `--windowsLogonAccount <account>` - Windows 사용자 이름을 형식 `domain\userName` 으로 지정하거나 지정하는 데 사용 `--runAsService --runAsAutoLogon` `userName@domain.com`
- `--windowsLogonPassword <password>` - Windows 로그인 암호를 사용 `--runAsService` 하거나 `--runAsAutoLogon` 지정하는 데 사용된다
(**그룹 관리 서비스 계정** 및 'NT AUTHORITY\NETWORK SERVICE'와 같은 Windows 기본 제공 계정에는 필요하지 않음)
- `--overwriteAutoLogon` - 컴퓨터에서 기존 자동 로그인을 덮어쓰는 데 사용됩니다 `--runAsAutoLogon` .
- `--noRestart` - 에이전트 구성이 완료된 후 호스트의 다시 시작을 중지하는 데 사용됩니다 `--runAsAutoLogon` .

배포 그룹만

- `--deploymentGroup` - 에이전트를 배포 그룹 에이전트로 구성
- `--deploymentGroupName <name>` - 에이전트가 `--deploymentGroup` 조인할 배포 그룹을 지정하는 데 사용됩니다.
- `--projectName <name>` - 프로젝트 이름을 설정하는 데 사용됩니다 `--deploymentGroup` .
- `--addDeploymentGroupTags` - 배포 그룹 태그를 추가해야 함을 나타내는 데 사용됩니다 `--deploymentGroup` .
- `--deploymentGroupTags <tags>` - 배포 그룹 에이전트에 대한 심표로 구분된 태그 목록을 지정하는 데 사용됩니다 `--addDeploymentGroupTags` (예: "web, db").

환경만

- `--addvirtualmachineresourcetags` - 환경 리소스 태그를 추가해야 함을 나타내는 데 사용됩니다.

- `--virtualmachineresourcetags <tags>` - 환경 리소스 에이전트에 대한 심표로 구분된 태그 목록을 지정하는 데 사용됩니다 `--addvirtualmachineresourcetags` (예: "web, db").

`./config.sh --help` 항상 최신 필수 및 선택적 응답을 나열합니다.

진단

자체 호스팅 에이전트에 문제가 있는 경우 진단을 실행해 볼 수 있습니다. 에이전트를 구성한 후:

```
./run.sh --diagnostics
```

이 기능은 문제 해결에 도움이 될 수 있는 진단 제품군을 통해 실행됩니다. 진단 기능은 에이전트 버전 2.165.0부터 사용할 수 있습니다.

다른 옵션에 대한 도움말

다른 옵션에 대해 알아보려면 다음을 수행합니다.

```
./config.sh --help
```

이 도움말은 인증 대안 및 무인 구성에 대한 정보를 제공합니다.

기능

에이전트의 기능은 처리할 수 있는 빌드 및 릴리스만 할당되도록 풀에서 카탈로그화되고 보급됩니다. [빌드 및 릴리스 에이전트 기능을 참조하세요.](#)

대부분의 경우 에이전트를 배포한 후에 소프트웨어 또는 유틸리티를 설치해야 합니다. 일반적으로 개발 머신에서 사용하는 소프트웨어와 도구가 무엇이든 에이전트에 설치해야 합니다.

예를 들어 빌드에 [npm 작업이 포함된 경우 npm](#)이 설치된 풀에 빌드 에이전트가 없는 한 빌드가 실행되지 않습니다.

중요

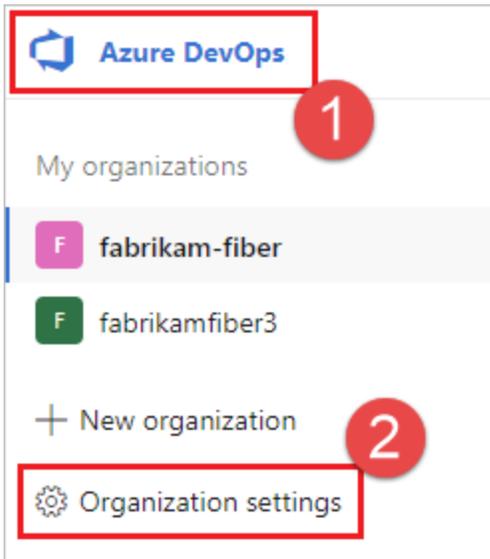
기능에는 모든 환경 변수와 에이전트가 실행되면 설정되는 값이 포함됩니다. 에이전트를 실행하는 동안 이러한 값이 변경되면 에이전트를 다시 시작하여 새 값을 선택해야 합니다. 에이전트에 새 소프트웨어를 설치한 후에는 빌드를 실행할 수 있도록 풀에 새 기능이 표시되도록 에이전트를 다시 시작해야 합니다.

환경 변수를 기능으로 제외하려는 경우 무시할 심표로 구분된 변수 목록을 사용하여 환경 변수 `VSO_AGENT_IGNORE` 를 설정하여 지정할 수 있습니다.

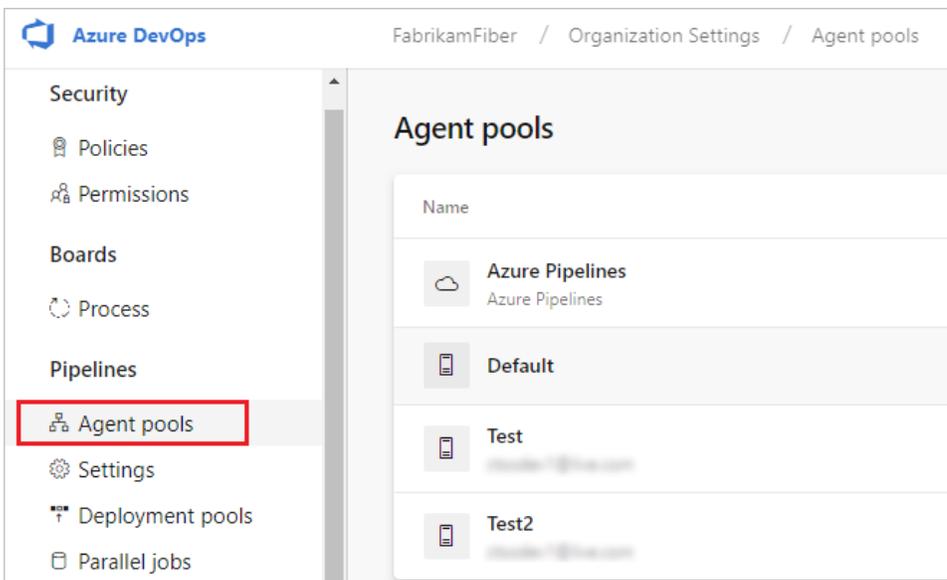
FAQ

최신 v2 에이전트 버전이 있는지 어떻게 할까요??

1. 에이전트 풀 탭으로 이동합니다.
 - a. Azure DevOps, 조직 설정을 선택합니다.

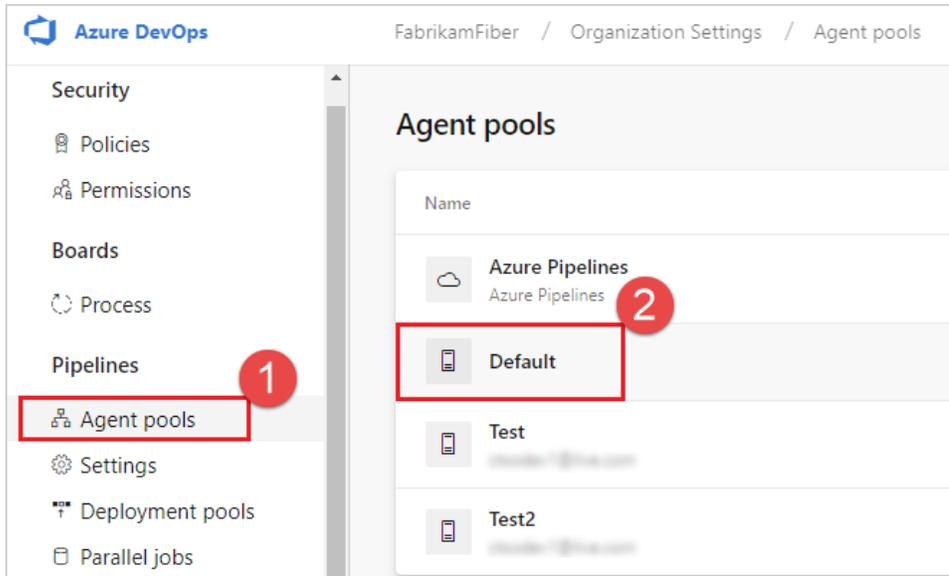


b. 에이전트 풀을 선택합니다.

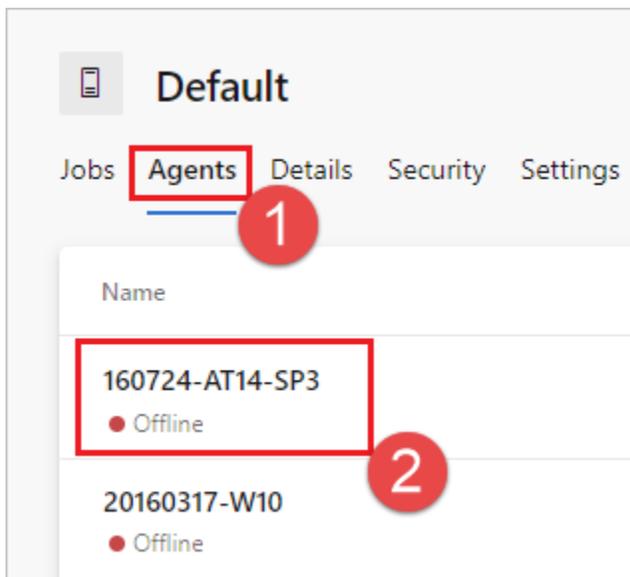


2. 에이전트가 포함된 풀을 클릭합니다.
3. 에이전트가 사용하도록 설정되어 있는지 확인합니다.
4. 기능 탭으로 이동합니다.

a. 에이전트 풀 탭에서 원하는 에이전트 풀을 선택합니다.



b. 에이전트를 선택하고 원하는 에이전트를 선택합니다.



c. 기능 탭을 선택합니다.

160724-AT14-SP3

Jobs **Capabilities**

User-defined capabilities +



No user-defined capabilities
Add a new capability

System capabilities Search by keyword

Name	Value
Agent.Name	160724-AT14-SP3
Agent.Version	2.107.0
Agent.ComputerName	160724-AT14-SP3

참고

Microsoft 호스팅 에이전트는 시스템 기능을 표시하지 않습니다. Microsoft 호스팅 에이전트에 설치된 소프트웨어 목록은 [Microsoft 호스팅 에이전트 사용](#)을 참조하세요.

5. Agent.Version 기능을 찾습니다. 최신 게시된 에이전트 버전에 대해 이 값을 확인할 수 있습니다. [Azure Pipelines 에이전트](#)를 참조하고 나열된 가장 높은 버전 번호에 대한 페이지를 확인합니다.
6. 각 에이전트는 최신 버전의 에이전트가 필요한 작업을 실행할 때 자동으로 업데이트됩니다. 일부 에이전트를 수동으로 업데이트하려면 풀을 마우스 오른쪽 단추로 클릭하고 **모든 에이전트 업데이트**를 선택합니다.

sudo가 서비스 명령을 실행하는 데 필요한 이유는 무엇인가요?

./svc.sh가 systemctl필요합니다.sudo

소스 코드: [GitHub의 systemd.svc.sh.template](#)

방화벽을 실행하고 있으며 코드가 Azure Repos에 있습니다. 에이전트는 어떤 URL로 통신해야 하나요?

방화벽 뒤의 보안 네트워크에서 에이전트를 실행하는 경우 에이전트가 다음 URL 및 IP 주소와의 통신을 시작할 수 있는지 확인합니다.

도메인 URL	Description
https://{organization_name}.pkgs.visualstudio.com	도메인을 사용하는 {organization_name}.visualstudio.com 조직을 위한 Azure DevOps 패키징 API
https://{organization_name}.visualstudio.com	도메인을 {organization_name}.visualstudio.com 사용하는 조직의 경우
https://{organization_name}.vsblob.visualstudio.com	도메인을 사용하는 {organization_name}.visualstudio.com 조직을 위한 Azure DevOps 원격 분석
https://{organization_name}.vsrm.visualstudio.com	도메인을 사용하는 {organization_name}.visualstudio.com 조직용 Release Management Services
https://{organization_name}.vssps.visualstudio.com	도메인을 사용하는 {organization_name}.visualstudio.com 조직을 위한 Azure DevOps Platform Services
https://{organization_name}.vstmr.visualstudio.com	도메인을 사용하는 {organization_name}.visualstudio.com 조직을 위한 Azure DevOps 테스트 관리 서비스
https://*.blob.core.windows.net	Azure Artifacts
https://*.dev.azure.com	도메인을 dev.azure.com 사용하는 조직의 경우

https://*.vsassets.io	CDN을 통한 Azure Artifacts
https://*.vsblob.visualstudio.com	도메인을 사용하는 dev.azure.com 조직을 위한 Azure DevOps 원격 분석
https://*.vssps.visualstudio.com	도메인을 사용하는 dev.azure.com 조직을 위한 Azure DevOps Platform Services
https://*.vstmr.visualstudio.com	도메인을 사용하는 dev.azure.com 조직을 위한 Azure DevOps 테스트 관리 서비스
https://app.vssps.visualstudio.com	도메인을 {organization_name}.visualstudio.com 사용하는 조직의 경우
https://dev.azure.com	도메인을 dev.azure.com 사용하는 조직의 경우
https://login.microsoftonline.com	Azure Active Directory 로그인
https://management.core.windows.net	Azure Management API
https://vstsagentpackage.azureedge.net	에이전트 패키지

조직이 기존 방화벽 또는 IP 제한 사항과 함께 작동하도록 하려면 IP 버전에 따라 다음 IP 주소를 포함하도록 허용 목록에 있는 IP가 열려 있는지 확인하고 dev.azure.com*dev.azure.com 업데이트합니다. 현재 허용 목록 13.107.6.183 인 경우 IP 13.107.9.183 주소를 제거할 필요가 없으므로 그대로 둡니다.

IPv4 범위

- 13.107.6.0/24
- 13.107.9.0/24
- 13.107.42.0/24
- 13.107.43.0/24

IPv6 범위

- 2620:1ec:4::/48
- 2620:1ec:a92::/48
- 2620:1ec:21::/48
- 2620:1ec:22::/48



참고

허용되는 주소에 대한 자세한 내용은 [허용된 주소 목록 및 네트워크 연결을 참조하세요](#).

자체 서명된 인증서를 사용하여 에이전트를 실행할 어떻게 할까요? 있나요?

[자체 서명된 인증서를 사용하여 에이전트 실행](#)

웹 프록시 뒤에서 에이전트를 실행할 어떻게 할까요? 있나요?

[웹 프록시 뒤에서 에이전트 실행](#)

에이전트를 다시 어떻게 할까요?

에이전트를 대화형으로 실행하는 경우 [대화형으로 실행](#)의 다시 시작 지침을 참조하세요. 에이전트를 시스템 서비스로 실행하는 경우 단계를 수행하여 [중지](#) 한 다음 에이전트를 [시작](#)합니다.

웹 프록시를 우회하고 Azure Pipelines에 연결하도록 에이전트를 어떻게 할까요? 구성하시겠습니까?

에이전트가 프록시를 우회하고 Azure Pipelines에 직접 연결하도록 하려면 에이전트가 다음 URL에 액세스할 수 있도록 웹 프록시를 구성해야 합니다.

도메인을 *.visualstudio.com 사용하는 조직의 경우:

```
https://login.microsoftonline.com
https://app.vssps.visualstudio.com
https://{organization_name}.visualstudio.com
https://{organization_name}.vsrm.visualstudio.com
https://{organization_name}.vstmr.visualstudio.com
https://{organization_name}.pkgs.visualstudio.com
https://{organization_name}.vssps.visualstudio.com
```

도메인을 dev.azure.com 사용하는 조직의 경우:

```

https://dev.azure.com
https://*.dev.azure.com
https://login.microsoftonline.com
https://management.core.windows.net
https://vstsagentpackage.azureedge.net
https://vssps.dev.azure.com

```

조직이 기존 방화벽 또는 IP 제한 사항과 함께 작동하도록 하려면 IP 버전에 따라 다음 IP 주소를 포함하도록 허용 목록에 있는 IP가 열려 있는지 확인하고 dev.azure.com*dev.azure.com 업데이트합니다. 현재 허용 목록 13.107.6.183 인 경우 IP 13.107.9.183 주소를 제거할 필요가 없으므로 그대로 둡니다.

IPv4 범위

- 13.107.6.0/24
- 13.107.9.0/24
- 13.107.42.0/24
- 13.107.43.0/24

IPv6 범위

- 2620:1ec:4::/48
- 2620:1ec:a92::/48
- 2620:1ec:21::/48
- 2620:1ec:22::/48

참고

이 절차를 통해 에이전트는 웹 프록시를 바이패스할 수 있습니다. 빌드 파이프라인 및 스크립트는 빌드에서 실행하는 각 작업 및 도구에 대한 웹 프록시 무시를 처리해야 합니다.

예를 들어 NuGet 작업을 사용하는 경우 사용 중인 NuGet 피드를 호스팅하는 서버의 URL 무시를 지원하도록 웹 프록시를 구성해야 합니다.

TFS를 사용하고 있으며 위의 섹션에 있는 URL이 작동하지 않습니다. 어디에서 도움을 받을 수 있나요?

[웹 사이트 설정 및 보안](#)

참고

- [Linux에서 Azure Pipelines 에이전트 배포 - Azure Pipelines | Microsoft Docs](#)