Kaynaklardan Kurulum

Wazuh yöneticisi ve aracısı, paketlerden kuruluma alternatif olarak kaynaklar aracılığıyla da kurulabilir.

- Wazuh Yöneticisini Kaynaklardan Yükleme
- Wazuh Aracısını Kaynaklardan Yükleme - Linux
- Wazuh Aracısını Kaynaklardan Yükleme - Windows
- Wazuh Aracısını Kaynaklardan Yükleme - macOS
- Wazuh Aracısını Kaynaklardan Yükleme - AIX
- Wazuh Aracısını Kaynaklardan Yükleme - HP-UX
- Wazuh Aracısını Kaynaklardan Yükleme - Solaris

Wazuh Yöneticisini Kaynaklardan Yükleme

Wazuh sunucusu, dağıtılan aracılardan alınan verileri toplar ve analiz eder. Wazuh yöneticisini, Wazuh API'sini ve Filebeat'i çalıştırır.

Wazuh yönetici paketi ve uyumlu ajan paketleri Paketler listesi bölümünden kontrol edilebilir veya indirilebilir .

Bağımlılıkları Yükleme

YUM

CentOS 7

```
yum update -y
yum install make gcc gcc-c++ policycoreutils-python automake autoconf libtool centos-release-scl openssl-devel wget bzip2 devtoolset-7 procps -y
curl -OL http://packages.wazuh.com/utils/gcc/gcc-9.4.0.tar.gz && tar xzf gcc-9.4.0.tar.gz  && cd gcc-9.4.0/ && ./contrib/download_prerequisites && ./configure --enable-languages=c,c++ --prefix=/usr --disable-multilib --disable-libsanitizer && make -j$(nproc) && make install && ln -fs /usr/bin/g++ /bin/c++ && ln -fs /usr/bin/gcc /bin/cc && cd .. && rm -rf gcc-* && scl enable devtoolset-7 bash
```

CMake 3.18 kurulumu.

```
curl -OL https://packages.wazuh.com/utils/cmake/cmake-3.18.3.tar.gz && tar -zxf cmake-3.18.3.tar.gz && cd cmake-3.18.3 && ./bootstrap --no-system-curl && make -j$(nproc) && make install
cd .. && rm -rf cmake-*
```

CentOS 8

```
yum install make cmake gcc gcc-c++ python3 python3-policycoreutils automake autoconf libtool openssl-devel yum-utils procps -y
curl -OL http://packages.wazuh.com/utils/gcc/gcc-9.4.0.tar.gz && tar xzf gcc-9.4.0.tar.gz  && cd gcc-9.4.0/ && ./contrib/download_prerequisites && ./configure --enable-languages=c,c++ --prefix=/usr --disable-multilib --
```

```
disable-libsanitizer && make -j$(nproc) && make install && ln -fs /usr/bin/g++ /bin/c++ && ln -fs /usr/bin/gcc /bin/cc && cd .. && rm -rf gcc-* && scl enable devtoolset-7 bash
yum-config-manager --enable powertools
yum install libstdc++-static -y
```

Kaynaklardan isteğe bağlı CMake 3.18 kurulumu

```
curl -OL https://packages.wazuh.com/utils/cmake/cmake-3.18.3.tar.gz && tar -zxf cmake-3.18.3.tar.gz && cd cmake-3.18.3 && ./bootstrap --no-system-curl && make -j$(nproc) && make install
cd .. && rm -rf cmake-*
export PATH=/usr/local/bin:$PATH
```

APT

```
apt-get update
apt-get install python gcc g++ make libc6-dev curl policycoreutils automake autoconf libtool libssl-dev procps
```

CMake 3.18 kurulumu.

```
curl -OL https://packages.wazuh.com/utils/cmake/cmake-3.18.3.tar.gz && tar -zxf cmake-3.18.3.tar.gz && cd cmake-3.18.3 && ./bootstrap --no-system-curl && make -j$(nproc) && make install
cd .. && rm -rf cmake-*
```

İsteğe bağlı: Aşağıdaki bağımlılıkları yalnızca CPython'u kaynaklardan derlerken yükleyin. v4.2.0'dan beri, yüklenmeye hazır taşınabilir bir CPython sürümü indirecektir. Yine de, 'yi çalıştırırken bayrağı ekleyerek CPython kaynaklarını indirebilirsiniz .`make deps TARGET=server PYTHON_SOURCE`make deps

Python yorumlayıcısını derlemek için gereken bağımlılıkları yüklemek için şu adımları izleyin:

Yum

```
yum install epel-release yum-utils -y
yum-builddep python34 -y
```

APT

```
echo "deb-src http://archive.ubuntu.com/ubuntu $(lsb_release -cs) main" >> /etc/apt/sources.list
apt-get update
apt-get build-dep python3 -y
```

> Not: Önceki komuttan alınan Python sürümü, ikili dosyaları derlemek için kullanılan işletim sistemine bağlı olarak değişebilir. Daha fazla bilgi için Bağımlılıkları yükleyin .

Wazuh yYöneticisinin Kurulumu

1. En son sürümü indirin ve çıkarın:

   ```
   curl -Ls https://github.com/wazuh/wazuh/archive/v4.9.2.tar.gz | tar zx
   cd wazuh-4.9.2
   ```

2. Daha önce başka bir platform için derleme yaptıysanız, Makefile'ı kullanarak derlemeyi temizleyin `src/`:

   ```
   make -C src clean
   make -C src clean-deps
   ```

3. Betiği çalıştırın `install.sh`. Bu, Wazuh kaynaklarını kullanarak kurulum sürecinde size rehberlik edecek bir sihirbaz görüntüler:

   > Uyarı: Veritabanı çıktısını etkinleştirmek istiyorsanız, kurulum betiğini çalıştırmadan önce bu bölümü inceleyin.

   ```
   ./install.sh
   ```

   İlk çalıştırma, güvenlik açığı algılama içeriğini indirip işlediği için biraz zaman alabilir . Bu süreci hızlandırmak için `DOWNLOAD_CONTENT`ortam değişkenini önceden olarak ayarlayabilirsiniz `y`. Ayarlanan komut, kurulum sırasında önceden hazırlanmış bir veritabanını indirir.

   ```
   DOWNLOAD_CONTENT=y ./install.sh
   ```

4. Script size ne tür bir kurulum istediğinizi sorduğunda `manager`Wazuh yöneticisini kurmak için şunu yazın:

```
1- What kind of installation do you want (manager, agent, local, hybrid, or help)? manager
```

> Not: Kurulum sırasında kullanıcılar kurulum yolunu belirleyebilir. Çalıştırın `./install.sh` ve dili seçin, kurulum modunu olarak ayarlayın `manager`, ardından kurulum yolunu ( ) ayarlayın. Varsayılan kurulum yolu 'dir . Yaygın olarak kullanılan özel bir yol . olabilir .`Choose where to install Wazuh [/var/ossec]`/var/ossec/opt

> Uyarı: Varsayılandan farklı bir yol seçerseniz kritik bir kurulum dizini seçmemeye son derece dikkat edin. Dizin zaten mevcutsa, yükleyici dizini silmenizi veya Wazuh'u içine kurarak devam etmenizi isteyecektir.

5. Kurulum programı kurulumun sonunda Wazuh'u başlatmak isteyip istemediğinizi sorar. Eğer istemezseniz, daha sonra şu şekilde başlatabilirsiniz:

   Systemd

   ```
   systemctl start wazuh-manager
   ```

   SysV Başlatma

   ```
   service wazuh-manager start
   ```

Diğer Wazuh Bileşenlerinin Kurulumu

Wazuh yöneticisi kaynaklardan yüklendikten sonra Kurulum kılavuzunu izleyerek Wazuh indeksleyicisini, Filebeat'i ve Wazuh panosunu yükleyebilirsiniz .

Kaldır

1. Wazuh yöneticisini kaldırmak için `WAZUH_HOME`geçerli kurulum yolunu ayarlayın:

   ```
   WAZUH_HOME="/WAZUH/INSTALLATION/PATH"
   ```

2. Hizmeti durdurun:

```
service wazuh-manager stop 2> /dev/null
```

3. Daemon'u durdurun:

```
$WAZUH_HOME/bin/wazuh-control stop 2> /dev/null
```

4. Kurulum klasörünü ve tüm içeriğini kaldırın:

```
rm -rf $WAZUH_HOME
```

5. Hizmeti silin:

Systemd

```
find /etc/systemd/system -name "wazuh*" | xargs rm -f
systemctl daemon-reload
```

SysV Başlatma

```
[ -f /etc/rc.local ] && sed -i'' '/wazuh-control start/d' /etc/rc.local
find /etc/{init.d,rc*.d} -name "*wazuh*" | xargs rm -f
```

6. Wazuh kullanıcı ve grubunu kaldır:

```
userdel wazuh 2> /dev/null
groupdel wazuh 2> /dev/null
```

Wazuh Aracısını Kaynaklardan Yükleme - Linux

> Not: Aşağıda açıklanan tüm komutları çalıştırmak için kök kullanıcı ayrıcalıklarına ihtiyacınız var. Wazuh 3.5'ten beri, bu işlemi takip ederken bir İnternet bağlantısına sahip olmak gerekir.

> Not: CMake 3.12.4, Wazuh aracı çözümünü derlemek için gereken en düşük kütüphane sürümüdür.

> Not: GCC 9.4, Wazuh aracı çözümünü derlemek için gereken en düşük derleyici sürümüdür.

1. Geliştirme araçlarını ve derleyicileri kurun. Linux'ta bu, dağıtımınızın paket yöneticisini kullanarak kolayca yapılabilir:

   Yum

   CentOS 6/7

   ```
   yum update -y
   yum install make gcc gcc-c++ policycoreutils-python automake autoconf libtool centos-release-scl openssl-devel wget bzip2 procps -y
   curl -OL http://packages.wazuh.com/utils/gcc/gcc-9.4.0.tar.gz && tar xzf gcc-9.4.0.tar.gz  && cd gcc-9.4.0/ && ./contrib/download_prerequisites && ./configure --enable-languages=c,c++ --prefix=/usr --disable-multilib --disable-libsanitizer && make -j$(nproc) && make install && ln -fs /bin/g++ /usr/bin/c++ && ln -fs /bin/gcc /usr/bin/cc && cd .. && rm -rf gcc-*
   ```

   CMake 3.18 kurulumu

```
curl -OL https://packages.wazuh.com/utils/cmake/cmake-3.18.3.tar.gz && tar -zxf cmake-3.18.3.tar.gz
cd cmake-3.18.3 && ./bootstrap --no-system-curl
make -j$(nproc) && make install
cd .. && rm -rf cmake-*
```

CentOS 8

```
yum install make gcc gcc-c++ python3 python3-policycoreutils automake autoconf libtool openssl-devel cmake procps -y
curl -OL http://packages.wazuh.com/utils/gcc/gcc-9.4.0.tar.gz && tar xzf gcc-9.4.0.tar.gz  && cd gcc-9.4.0/ && ./contrib/download_prerequisites && ./configure --enable-languages=c,c++ --prefix=/usr --disable-multilib --disable-libsanitizer && make -j$(nproc) && make install && ln -fs /bin/g++ /usr/bin/c++ && ln -fs /bin/gcc /usr/bin/cc && cd .. && rm -rf gcc-*
yum-config-manager --enable powertools
yum install libstdc++-static -y
```

CMake 3.18 kurulumu

```
curl -OL https://packages.wazuh.com/utils/cmake/cmake-3.18.3.tar.gz && tar -zxf cmake-3.18.3.tar.gz && cd cmake-3.18.3 && ./bootstrap --no-system-curl && make -j$(nproc) && make install
cd .. && rm -rf cmake-*
export PATH=/usr/local/bin:$PATH
```

APT

```
apt-get install python gcc g++ make libc6-dev curl policycoreutils automake autoconf libtool libssl-dev procps
```

CMake 3.18 kurulumu

```
curl -OL https://packages.wazuh.com/utils/cmake/cmake-3.18.3.tar.gz && tar -zxf cmake-3.18.3.tar.gz && cd cmake-3.18.3 && ./bootstrap --no-system-curl && make -j$(nproc) && make install
cd .. && rm -rf cmake-*
```

ZYpp

```
zypper install -y make gcc gcc-c++ policycoreutils-python automake autoconf libtool libopenssl-devel curl
```

CMake 3.18 kurulumu

```
curl -OL https://packages.wazuh.com/utils/cmake/cmake-3.18.3.tar.gz && tar -zxf cmake-3.18.3.tar.gz
&& cd cmake-3.18.3 && ./bootstrap --no-system-curl && make -j$(nproc) && make install
cd .. && rm -rf cmake-*
```

> Not: Suse 11 için bazı araçların paket yöneticisinde bulunmaması mümkün olabilir, bu durumda aşağıdaki resmi depoları ekleyebilirsiniz:

```
zypper addrepo http://download.opensuse.org/distribution/11.4/repo/oss/ oss
```

Pacman

Wazuh'u derlemek için önerilen sürüm GCC/G++ 9.4'tür.

```
pacman --noconfirm -Syu curl gcc make sudo wget expect gnupg perl-base perl fakeroot python brotli
automa
```

2. En son sürümü indirin ve çıkarın:

```
curl -Ls https://github.com/wazuh/wazuh/archive/v4.9.2.tar.gz | tar zx
```

3. Betiği çalıştırın `install.sh`. Bu, Wazuh kaynaklarını kullanarak kurulum sürecinde size rehberlik edecek bir sihirbazı çalıştıracaktır:

```
cd wazuh-4.9.2
./install.sh
```

Daha önce başka bir platform için derleme yaptıysanız, Makefile'ı kullanarak derlemeyi temizlemelisiniz `src`:

```
cd wazuh-4.9.2
make -C src clean
make -C src clean-deps
```

> Not: Kurulum sırasında kullanıcılar kurulum yolunu belirleyebilir. Çalıştırın `./install.sh` ve dili seçin, kurulum modunu olarak ayarlayın `agent`, ardından kurulum yolunu (. `Choose where to install Wazuh [/var/ossec]`) ayarlayın. Varsayılan kurulum yolu `/var/ossec` 'dir . Yaygın olarak kullanılan özel bir yol `/opt` olabilir . Varsayılandan farklı bir yol

> seçerken, dizin zaten mevcutsa, yükleyici dizini silmeyi veya içine Wazuh'u yüklemeyi isteyecektir. Ayrıca gözetimsiz bir kurulum da çalıştırabilirsiniz.

4. Komut dosyası ne tür bir kurulum istediğinizi soracaktır. Wazuh aracısını kurmak için aracı yazın:

Output

```
1- What kind of installation do you want (manager, agent, local, hybrid or help)? agent
```

Sonraki adımlar

Artık aracı yüklendiğine göre, bir sonraki adım aracı Wazuh sunucusuna kaydetmektir. Bu süreç hakkında daha fazla bilgi için lütfen Wazuh aracı kayıt bölümünü kontrol edin.

Kaldır

Wazuh aracısını kaldırmak için WAZUH_HOME'u geçerli yükleme yoluyla ayarlayın:

```
WAZUH_HOME="/WAZUH/INSTALLATION/PATH"
```

Hizmeti durdurun:

```
service wazuh-agent stop 2> /dev/null
```

Daemon'u durdurun:

```
$WAZUH_HOME/bin/wazuh-control stop 2> /dev/null
```

Kurulum klasörünü ve tüm içeriğini kaldırın:

```
rm -rf $WAZUH_HOME
```

Hizmeti silin:

SysV Başlatma

```
[ -f /etc/rc.local ] && sed -i'' '/wazuh-control start/d' /etc/rc.local
find /etc/{init.d,rc*.d} -name "*wazuh*" | xargs rm -f
```

Systemd

```
find /etc/systemd/system -name "wazuh*" | xargs rm -f
systemctl daemon-reload
```

Wazuh kullanıcı ve grubunu kaldır:

```
userdel wazuh 2> /dev/null
groupdel wazuh 2> /dev/null
```

curl -OL https://packages.wazuh.com/utils/cmake/cmake-3.18.3.tar.gz && tar -zxf cmake-3.18.3.tar.gz && cd cmake-3.18.3 && ./bootstrap --no-system-curl && make -j$(nproc) && make install
cd .. && rm -rf cmake-*
export PATH=/usr/local/bin:$PATH

Wazuh Aracısını Kaynaklardan Yükleme - Windows

> Not: Aşağıdaki prosedür Ubuntu 22.04'te test edilmiştir ve diğer Debian/Ubuntu sürümlerinde de çalışabilir. MinGW 10 kullanmak gerekir.

1. Ubuntu yapı ortamını kurun. Ubuntu'da Windows Wazuh aracı yükleyicisini oluşturmak için bu bağımlılıkları yükleyin:

   ```
   apt-get install curl gcc-mingw-w64 g++-mingw-w64-i686 g++-mingw-w64-x86-64 nsis make cmake
   ```

2. Windows yapı ortamını kurun. Yükleyiciyi oluşturmak için, Windows makinesinde aşağıdaki bağımlılıkların bulunması gerekir:

   - WiX Araç Seti .
   - .NET çerçevesi 3.5.1.
   - Microsoft Windows SDK'sı.

3. Wazuh kaynak kodunu Ubuntu makinesine indirin ve sıkıştırılmış dosyayı açın:

   ```
   curl -Ls https://github.com/wazuh/wazuh/archive/v4.9.2.tar.gz | tar zx
   cd wazuh-4.9.2/src
   ```

4. Aşağıdaki komutu çalıştırarak Agent'ı derleyin `make`:

   ```
   make deps TARGET=winagent
   make TARGET=winagent
   ```

   İnşa sürecinin sonunda aşağıdaki çıktı görünecektir:

   Output

   ```
   Done building winagent
   ```

5. Tüm depoyu Windows makinesine taşır. İşlemi hızlandırmak için sıkıştırılması önerilir.

```
cd ../.. && zip -r wazuh.zip wazuh-4.9.2
```

6. Depoyu Windows makinesinde açın, `wazuh-installer-build-msi.bat`betiği win32 klasöründen çalıştırın.

```
cd wazuh-4.9.2\src\win32
.\wazuh-installer-build-msi.bat
```

Eğer yükleyiciyi imzalamak istemiyorsanız, önceki scriptte signtool satırını yorum olarak yazmanız veya silmeniz gerekecektir.

```
:: signtool sign /a /tr http://timestamp.digicert.com /d "%MSI_NAME%" /fd SHA256 /td SHA256
"%MSI_NAME%"
```

7. İstendiğinde sürümü ve revizyon numarasını belirtin. Bu aynı zamanda Windows yükleyici dosyasını da oluşturacaktır. Aşağıdaki çıktıda sürüm 4.9.2 olarak ayarlanmıştır ve revizyon 1 olarak ayarlanmıştır. Bu, Windows yükleyicisini oluşturur`wazuh-agent-4.9.2-1.msi`

```
C:\wazuh\wazuh-4.9.2\src\win32>REM IF VERSION or REVISION are empty, ask for their value

C:\wazuh\wazuh-4.9.2\src\win32>IF [] == [] set /p VERSION=Enter the version of the Wazuh agent (x.y.z):
Enter the version of the Wazuh agent (x.y.z):4.9.2

C:\wazuh\wazuh-4.9.2\src\win32>IF [] == [] set /p REVISION=Enter the revision of the Wazuh agent:
Enter the revision of the Wazuh agent:1

C:\wazuh\wazuh-4.9.2\src\win32>SET MSI_NAME=wazuh-agent-4.9.2-1.msi
```

8. Kurulum kılavuzunu izleyerek wazuh-agent-4.9.2-1.msi kurulumunu gerçekleştirin .

Sonraki adımlar

Artık aracı yüklendiğine göre, bir sonraki adım aracı Wazuh sunucusuna kaydetmektir. Bu süreç hakkında daha fazla bilgi için lütfen Wazuh aracı kayıt bölümünü kontrol edin.

Kaldır

Aracı kaldırmak için, gözetimsiz işlemi gerçekleştirmek üzere orijinal MSI dosyasına ihtiyaç duyulacaktır:

```
msiexec.exe /x wazuh-agent-4.9.2-1.msi /qn
```

Wazuh Aracısını Kaynaklardan Yükleme - macOS

1. macOS için bir paket yöneticisi olan brew'u kurun:

```
$ /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

> Uyarı: Bazı macOS sürümlerinde, bu ilk komut *homebrew/core*'*un* sığ klon olduğunu bildirerek başarısız olabilir. Bu olursa, sorunu çözmek için aşağıdaki komutları çalıştırın.

```
rm -rf "/usr/local/Homebrew/Library/Taps/homebrew/homebrew-core"
brew tap homebrew/core
```

> Daha sonra brew'u düzgün bir şekilde kurmak için ilkini tekrar çalıştırın: /bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

2. Geliştirme araçlarını ve derleyicileri brew aracılığıyla kurun.

```
brew install automake autoconf libtool cmake
```

3. En son sürümü indirin ve çıkarın:

```
curl -Ls https://github.com/wazuh/wazuh/archive/v4.9.2.tar.gz | tar zx
```

> Not: Aşağıda anlatılan tüm komutların root kullanıcı ayrıcalıklarıyla yürütülmesi gerekmektedir.

4. Betiği çalıştırın `install.sh`. Bu, Wazuh kaynaklarını kullanarak kurulum sürecinde size rehberlik edecek bir sihirbazı çalıştıracaktır:

```
cd wazuh-4.9.2
USER_DIR="/Library/Ossec" ./install.sh
```

> Not: Değişkenle birlikte `USER_DIR`, aracı yükleme yolunun belirtildiğini unutmayın `/Library/Ossec`

Daha önce başka bir platform için derleme yaptıysanız, Makefile'ı kullanarak derlemeyi temizlemelisiniz `src`:

```
cd wazuh-4.9.2
make -C src clean
make -C src clean-deps
```

> Not: Kurulum sırasında kullanıcılar kurulum yolunu belirleyebilir. Çalıştırın `./install.sh` ve dili seçin, kurulum modunu olarak ayarlayın `agent`, ardından kurulum yolunu ( ) ayarlayın. Varsayılan kurulum yolu 'dir . Varsayılandan farklı bir yol seçerken, dizin zaten mevcutsa, yükleyici dizini silmenizi veya Wazuh'u içine kurarak devam etmenizi isteyecektir. Ayrıca gözetimsiz bir kurulum da çalıştırabilirsiniz .`Choose where to install Wazuh [/Library/Ossec]/Library/Ossec`

5. Komut dosyası ne tür bir kurulum istediğinizi soracaktır. Wazuh aracısını kurmak için aracı yazın:

Output

```
1- What kind of installation do you want (manager, agent, local, hybrid, or help)? agent
```

Sonraki adımlar

Artık aracı yüklendiğine göre, bir sonraki adım aracı Wazuh sunucusuna kaydetmektir. Bu süreç hakkında daha fazla bilgi için lütfen Wazuh aracı kayıt bölümünü kontrol edin.

Kaldır

Wazuh aracısını kaldırmak için `WAZUH_HOME`geçerli kurulum yolunu ayarlayın:

```
WAZUH_HOME="/WAZUH/INSTALLATION/PATH"
```

Hizmeti durdurun:

```
service wazuh-agent stop 2> /dev/null
```

Daemon'u durdurun:

```
$WAZUH_HOME/bin/wazuh-control stop 2> /dev/null
```

Kurulum klasörünü ve tüm içeriğini kaldırın:

```
rm -rf $WAZUH_HOME
```

Hizmeti silin:

```
rm -rf /Library/StartupItems/WAZUH
```

Wazuh kullanıcı ve grubunu kaldır:

```
dscl . -delete "/Users/wazuh" > /dev/null 2>&1
dscl . -delete "/Groups/wazuh" > /dev/null 2>&1
```

Wazuh Aracısını Kaynaklardan Yükleme - AIX

Aşağıdaki kurulum prosedürü için desteklenen sürüm AIX 6.1 TL9 veya üzeridir.

> Not: Aşağıda açıklanan tüm komutların root kullanıcı ayrıcalıklarıyla yürütülmesi gerekir. Wazuh 3.5'ten beri, bu işlemi takip ederken bir İnternet bağlantısına sahip olmak gerekir.

1. Geliştirme araçlarını ve derleyicileri kurun.
   1. Aracı indirin `wget`.

      ```
      rpm -Uvh --nodeps http://packages-dev.wazuh.com/deps/aix/wget-1.19-1.aix6.1.ppc.rpm
      ```

   2. Aşağıdaki betiği indirin.

      ```
      wget https://raw.githubusercontent.com/wazuh/wazuh-packages/4.9/aix/generate_wazuh_packages.sh --no-check-certificate
      ```

      > Not: Eğer scripti bu şekilde indiremiyorsanız, o zaman başka bir makine kullanarak indirip scp yardımcı programını kullanarak AIX makinesine kopyalamalısınız.

   3. Bash ve libiconv'u indirin.

      ```
      rpm -Uvh --nodeps http://packages-dev.wazuh.com/deps/aix/bash-4.4-4.aix6.1.ppc.rpm
      rpm -Uvh --nodeps http://packages-dev.wazuh.com/deps/aix/libiconv-1.14-22.aix6.1.ppc.rpm
      ```

   4. Gerekli bağımlılıkları scripti kullanarak kurun.

      ```
      chmod +x generate_wazuh_packages.sh
      ./generate_wazuh_packages.sh -e
      ```

      > Not: Bu adım birkaç dakika sürebilir.

2. En son sürümü indirin.

```
wget -O wazuh.tar.gz --no-check-certificate https://api.github.com/repos/wazuh/wazuh/tarball/v4.9.2
gunzip -c wazuh.tar.gz | tar -xvf -
```

> Not: Eğer deponuzu bu şekilde indiremiyorsanız scp yardımcı programını kullanarak kopyalamalısınız.

3. Kaynakları derleyin.

```
cd wazuh-4.9.2
cd src
gmake clean-deps
gmake clean
gmake deps TARGET=agent RESOURCES_URL=http://packages.wazuh.com/deps/27
gmake TARGET=agent USE_SELINUX=no PREFIX=/var/ossec
```

4. Betiği çalıştırın `install.sh`. Bu, Wazuh kaynaklarını kullanarak kurulum sürecinde size rehberlik edecek bir sihirbazı çalıştıracaktır:

```
cd ..
./install.sh
```

Daha önce başka bir platform için derleme yaptıysanız, Makefile'ı kullanarak derlemeyi temizlemelisiniz `src`:

```
gmake -C src clean-deps
gmake -C src clean
```

> Not: Kurulum sırasında kullanıcılar kurulum yolunu belirleyebilir. Çalıştırın `./install.sh` ve dili seçin, kurulum modunu olarak ayarlayın `agent`, ardından kurulum yolunu ( ) ayarlayın. Varsayılan kurulum yolu 'dir . Yaygın olarak kullanılan özel bir yol . olabilir . Varsayılandan farklı bir yol seçerken, dizin zaten mevcutsa, yükleyici dizini silmeyi veya Wazuh'u içine yüklemeyi isteyecektir. Ayrıca gözetimsiz bir kurulum da çalıştırabilirsiniz .`Choose where to install Wazuh [/var/ossec]`/var/ossec/opt

5. Son olarak aşağıdaki yapılandırmayı uygulayın:

```
sed '/System inventory/,/^$/{/^$/!d;}' /var/ossec/etc/ossec.conf > /var/ossec/etc/ossec.conf.tmp
mv /var/ossec/etc/ossec.conf.tmp /var/ossec/etc/ossec.conf
```

> Not: Yukarıdaki komutların varsayılan kurulum yolu için yürütüldüğünü unutmayın `/var/ossec`. Aracı başka bir yola kurduysanız, bu komutların yolunu değiştirmeniz gerekecektir.

Sonraki adımlar

Artık aracı yüklendiğine göre, bir sonraki adım aracı Wazuh sunucusuna kaydetmektir. Bu süreç hakkında daha fazla bilgi için lütfen Wazuh aracı kayıt bölümünü kontrol edin.

Kaldır

Wazuh aracısını kaldırmak için `WAZUH_HOME`geçerli kurulum yolunu ayarlayın:

```
WAZUH_HOME="/WAZUH/INSTALLATION/PATH"
```

Hizmeti durdurun:

```
service wazuh-agent stop 2> /dev/null
```

Daemon'u durdurun:

```
$WAZUH_HOME/bin/wazuh-control stop 2> /dev/null
```

Kurulum klasörünü ve tüm içeriğini kaldırın:

```
rm -rf $WAZUH_HOME
```

Hizmeti silin:

```
find /etc/rc.d -name "*wazuh*" | xargs rm -f
```

Wazuh kullanıcı ve grubunu kaldır:

```
userdel wazuh 2> /dev/null
groupdel wazuh 2> /dev/null
```

Wazuh Aracısını Kaynaklardan Yükleme - HP-UX

> Not: Aşağıda açıklanan tüm komutların root kullanıcı ayrıcalıklarıyla yürütülmesi gerekir. Wazuh 3.5'ten beri, bu işlemi takip ederken bir İnternet bağlantısına sahip olmak gerekir.

1. Geliştirme araçlarını ve derleyicileri kurun.
    1. Dosyayı indirin `depothelper-2.10-hppa_32-11.31.depot`.

        ```
        /usr/local/bin/wget https://github.com/wazuh/wazuh-packages/raw/master/hp-ux/depothelper-2.10-hppa_32-11.31.depot --no-check-certificate
        ```

        > Not: Eğer scripti bu şekilde indiremiyorsanız, o zaman başka bir makine kullanarak indirip scp yardımcı programını kullanarak HP-UX makinesine kopyalamalısınız.

    2. Paket yöneticisini yükleyin. Depo dosyasına giden mutlak yol kullanılır.

        ```
        swinstall -s /ABSOLUTE/PATH/depothelper-2.10-hppa_32-11.31.depot \*
        ```

    3. Aracı indirin `wget`(Eğer yüklü değilse).

        ```
        /usr/local/bin/depothelper -f wget
        ```

    4. Aşağıdaki betiği indirin

        ```
        /usr/local/bin/wget https://raw.githubusercontent.com/wazuh/wazuh-packages/master/hp-ux/generate_wazuh_packages.sh --no-check-certificate
        ```

> Not: Eğer scripti bu şekilde indiremiyorsanız scp yardımcı programını kullanarak kopyalamalısınız.

5. Gerekli bağımlılıkları scripti kullanarak kurun.

```
chmod +x generate_wazuh_packages.sh
./generate_wazuh_packages.sh -e
```

> Not: Bu adım uzun sürebilir.

2. En son sürümü indirin.

```
/usr/local/bin/curl -k -L -O https://github.com/wazuh/wazuh/archive/v4.9.2.zip && /usr/local/bin/unzip v4.9.2
```

> Not: Eğer deponuzu bu şekilde indiremiyorsanız scp yardımcı programını kullanarak kopyalamalısınız.

3. Kaynakları derleyin.

```
cd wazuh-4.9.2
/usr/local/bin/gmake -C src deps RESOURCES_URL=http://packages.wazuh.com/deps/27 TARGET=agent
/usr/local/bin/gmake -C src TARGET=agent USE_SELINUX=no
```

4. Betiği çalıştırın `install.sh`. Bu, Wazuh kaynaklarını kullanarak kurulum sürecinde size rehberlik edecek bir sihirbazı çalıştıracaktır:

```
./install.sh
```

Daha önce başka bir platform için derleme yaptıysanız, Makefile'ı kullanarak derlemeyi temizlemelisiniz `src`:

```
/usr/local/bin/gmake -C src clean-deps
/usr/local/bin/gmake -C src clean
```

> Not: Kurulum sırasında kullanıcılar kurulum yolunu belirleyebilir. Çalıştırın `./install.sh` ve dili seçin, kurulum modunu olarak ayarlayın `agent`, ardından kurulum yolunu ( ) ayarlayın. Varsayılan kurulum yolu 'dir . Yaygın olarak kullanılan özel bir yol . olabilir . Varsayılandan farklı bir yol seçerken, dizin zaten mevcutsa, yükleyici dizini silmeyi veya Wazuh'u içine yüklemeyi isteyecektir. Ayrıca

> gözetimsiz bir kurulum da çalıştırabilirsiniz .`Choose where to install Wazuh [/var/ossec]` `/var/ossec/opt`

Sonraki adımlar

Artık aracı yüklendiğine göre, bir sonraki adım aracı Wazuh sunucusuna kaydetmektir. Bu süreç hakkında daha fazla bilgi için lütfen Wazuh aracı kayıt bölümünü kontrol edin.

Kaldır

Wazuh aracısını kaldırmak için `WAZUH_HOME`geçerli kurulum yolunu ayarlayın:

```
WAZUH_HOME="/WAZUH/INSTALLATION/PATH"
```

Hizmeti durdurun:

```
service wazuh-agent stop 2> /dev/null
```

Daemon'u durdurun:

```
$WAZUH_HOME/bin/wazuh-control stop 2> /dev/null
```

Kurulum klasörünü ve tüm içeriğini kaldırın:

```
rm -rf $WAZUH_HOME
```

Hizmeti silin:

```
find /sbin/{init.d,rc*.d} -name "*wazuh*" | xargs rm -f
```

Wazuh kullanıcı ve grubunu kaldır:

```
userdel wazuh 2> /dev/null
groupdel wazuh 2> /dev/null
```

Wazuh Aracısını Kaynaklardan Yükleme - Solaris

Bu bölümde, aşağıdaki Solaris sürümleri için Wazuh aracısının kaynaklardan nasıl indirileceği ve oluşturulacağı açıklanmaktadır:

- Solaris i386 için
- Solaris SPARC için

Solaris 10

> Not: Aşağıda açıklanan tüm komutların root kullanıcı ayrıcalıklarıyla yürütülmesi gerekir. Wazuh 3.5'ten beri, bu işlemi takip ederken bir İnternet bağlantısına sahip olmak gerekir.

1. Geliştirme araçlarını ve derleyicileri kurun.

    1. Bash kabuğunu çalıştırın ve pkgutil'i yükleyin.

        ```
        bash
        PATH="${PATH}:/usr/sbin:/usr/bin:/usr/sbin/:/opt/csw/gnu/:/usr/sfw/bin/:/opt/csw/bin/"
        export PATH
        pkgadd -d http://get.opencsw.org/now
        ```

    2. Aşağıdaki araçları yükleyin:

        ```
        /opt/csw/bin/pkgutil -y -i git cmake automake autoconf gmake libtool wget curl gcc5core
        gcc5g++ gtar
        ```

    3. gcc/g++ 5.5 derleyicisini indirin ve derleyin:

```
curl -L http://packages.wazuh.com/utils/gcc/gcc-5.5.0.tar.gz | gtar xz && cd gcc-5.5.0
curl -L http://packages.wazuh.com/utils/gcc/mpfr-2.4.2.tar.bz2 | gtar xj && mv mpfr-2.4.2 mpfr
curl -L http://packages.wazuh.com/utils/gcc/gmp-4.3.2.tar.bz2 | gtar xj && mv gmp-4.3.2 gmp
curl -L http://packages.wazuh.com/utils/gcc/mpc-0.8.1.tar.gz | gtar xz && mv mpc-0.8.1 mpc
curl -L http://packages.wazuh.com/utils/gcc/isl-0.14.tar.bz2 | gtar xj && mv isl-0.14 isl
unset CPLUS_INCLUDE_PATH && unset LD_LIBRARY_PATH
export PATH=/usr/sbin:/usr/bin:/usr/ccs/bin:/opt/csw/bin
mkdir -p /usr/local
./configure --prefix=/usr/local/gcc-5.5.0 --enable-languages=c,c++ --disable-multilib --disable-libsanitizer --disable-bootstrap --with-ld=/usr/ccs/bin/ld --without-gnu-ld --with-gnu-as --with-as=/opt/csw/bin/gas
gmake && gmake install
export CPLUS_INCLUDE_PATH=/usr/local/gcc-5.5.0/include/c++/5.5.0
export LD_LIBRARY_PATH=/usr/local/gcc-5.5.0/lib
echo "export PATH=/usr/sbin:/usr/bin:/usr/ccs/bin:/opt/csw/bin" >> /etc/profile
echo "export CPLUS_INCLUDE_PATH=/usr/local/gcc-5.5.0/include/c++/5.5.0" >> /etc/profile
echo "export LD_LIBRARY_PATH=/usr/local/gcc-5.5.0/lib" >> /etc/profile
rm -rf gcc-*
ln -sf /usr/local/gcc-5.5.0/bin/g++ /usr/bin/g++
cd ..
```

> Not: Adımın `gmake` tamamlanması birkaç dakika sürecektir. Bu normal bir davranıştır.

4. cmake kütüphanesini kurun:

```
curl -sL http://packages.wazuh.com/utils/cmake/cmake-3.18.3.tar.gz | gtar xz
cd cmake-3.18.3
./bootstrap
gmake && gmake install
cd .. && rm -rf cmake-3.18.3
ln -sf /usr/local/bin/cmake /usr/bin/cmake
```

5. Perl 5.10.1'i indirin ve kurun.

```
wget http://www.cpan.org/src/5.0/perl-5.10.1.tar.gz
gunzip ./perl-5.10.1.tar.gz && tar xvf perl-5.10.1.tar
cd perl-5.10.1
./Configure -Dcc=gcc -d -e -s
gmake clean && gmake -d -s
gmake install -d -s
cd ..
```

6. Perl'in eski sürümünü kaldırıp yerine perl 5.10.1 sürümünü yükleyin.

```
rm /usr/bin/perl
mv /opt/csw/bin/perl5.10.1 /usr/bin/
mv /usr/bin/perl5.10.1 /usr/bin/perl
rm -rf perl-5.10.1*
```

2. Wazuh'un son sürümünü indirin.

```
/opt/csw/bin/git clone -b v4.9.2 https://github.com/wazuh/wazuh.git
```

3. Kaynakları derleyin.

- Solaris 10 i386 için:

```
export PATH=/usr/local/gcc-5.5.0/bin:/usr/sbin:/usr/bin:/usr/ccs/bin:/opt/csw/bin:/opt/csw/gnu
export CPLUS_INCLUDE_PATH=/usr/local/gcc-5.5.0/include/c++/5.5.0
export LD_LIBRARY_PATH=/usr/local/gcc-5.5.0/lib
cd wazuh/src
gmake clean
gmake deps TARGET=agent
gmake -j 4 TARGET=agent PREFIX=/var/ossec USE_SELINUX=no
cd ..
```

- Solaris 10 SPARC için:

```
export PATH=/usr/local/gcc-5.5.0/bin:/usr/sbin:/usr/bin:/usr/ccs/bin:/opt/csw/bin:/opt/csw/gnu
export CPLUS_INCLUDE_PATH=/usr/local/gcc-5.5.0/include/c++/5.5.0
export LD_LIBRARY_PATH=/usr/local/gcc-5.5.0/lib
cd wazuh/src
gmake clean
gmake deps TARGET=agent
gmake -j 4 TARGET=agent PREFIX=/var/ossec USE_SELINUX=no
```

```
cd ..
```

4. Shebang'i değiştirmek için solaris 10 sh dosyalarını yamalayın.

```
for file in $(find . -name "*.sh");do
sed 's:#!/bin/sh:#!/usr/xpg4/bin/sh:g' $file > $file.new
mv $file.new $file && chmod +x $file
done
```

5. Betiği çalıştırın `install.sh`. Bu, Wazuh kaynaklarını kullanarak kurulum sürecinde size rehberlik edecek bir sihirbazı çalıştıracaktır:

```
bash install.sh
```

Daha önce başka bir platform için derleme yaptıysanız, src'deki Makefile'ı kullanarak derlemeyi temizlemelisiniz:

```
gmake -C src clean
gmake -C src clean-deps
```

> Not: Kurulum sırasında kullanıcılar kurulum yolunu belirleyebilir. Çalıştırın `./install.sh` ve dili seçin, kurulum modunu olarak ayarlayın `agent`, ardından kurulum yolunu (`Choose where to install Wazuh [/var/ossec]`) ayarlayın. Varsayılan kurulum yolu `/var/ossec` 'dir . Yaygın olarak kullanılan özel bir yol `/opt` olabilir . Varsayılandan farklı bir yol seçerken, dizin zaten mevcutsa, yükleyici dizini silmeyi veya içine Wazuh'u yüklemeyi isteyecektir. Ayrıca gözetimsiz bir kurulum da çalıştırabilirsiniz .

6. Komut dosyası ne tür bir kurulum istediğinizi soracaktır. `agent`Wazuh aracısını kurmak için şunu yazın:

Output

```
1- What kind of installation do you want (manager, agent, local, hybrid, or help)? agent
```

Sonraki adımlar

Artık aracı yüklendiğine göre, bir sonraki adım aracı Wazuh sunucusuna kaydetmektir. Bu süreç hakkında daha fazla bilgi için lütfen Wazuh aracı kayıt bölümünü kontrol edin.

Kaldır

Wazuh aracısını kaldırmak için `WAZUH_HOME` geçerli kurulum yolunu ayarlayın:

```
WAZUH_HOME="/WAZUH/INSTALLATION/PATH"
```

Hizmeti durdurun:

```
service wazuh-agent stop 2> /dev/null
```

Daemon'u durdurun:

```
$WAZUH_HOME/bin/wazuh-control stop 2> /dev/null
```

Kurulum klasörünü ve tüm içeriğini kaldırın:

```
rm -rf $WAZUH_HOME
```

Hizmeti silin:

```
find /etc/{init.d,rc*.d} -name "*wazuh*" | xargs rm -f
```

Wazuh kullanıcı ve grubunu kaldır:

```
userdel wazuh 2> /dev/null
groupdel wazuh 2> /dev/null
```

Solaris 11

> Not: Aşağıda açıklanan tüm komutların root kullanıcı ayrıcalıklarıyla yürütülmesi gerekir. Wazuh 3.5'ten beri, bu işlemi takip ederken bir İnternet bağlantısına sahip olmak gerekir.

Geliştirme araçlarını ve derleyicileri kurun.

1. Pkgutil'i kurun ve güncelleyin.

```
pkgadd -d http://get.opencsw.org/now
export PATH="${PATH}:/usr/sfw/bin:/opt/csw/bin:/opt/ccs/bin"
pkgutil -y -U
```

2. Python 2.7'yi kurun.

```
/opt/csw/bin/pkgutil -y -i python27
ln -sf /opt/csw/bin/python2.7 /usr/bin/python
```

3. Aşağıdaki araçları yükleyin:

```
/opt/csw/bin/pkgutil -y -i git gmake cmake gcc5core gcc5g++
```

4. Bir sonraki adımda ihtiyaç duyulacak tüm dosyaları içerecek şekilde bir gcc sürümü yükleyin:

```
# pkg install gcc-45
```

5. gcc/g++ 5.5 derleyicisini indirin ve derleyin:

```
curl -O https://packages.wazuh.com/utils/gcc/gcc-5.5.0.tar.gz && gtar xzf gcc-5.5.0.tar.gz
ln -sf gcc-5.5.0 gcc
cd gcc && ./contrib/download_prerequisites
cd .. && mkdir -p gcc-build && cd gcc-build
../gcc/configure --prefix=/usr/local/gcc-5.5.0 --enable-languages=c,c++ --disable-multilib --
disable-libsanitizer --disable-bootstrap --with-ld=/usr/ccs/bin/ld --without-gnu-ld --with-gnu-as --
with-as=/opt/csw/bin/gas
gmake
gmake install
export PATH=/usr/local/gcc-5.5.0/bin/:/usr/local/bin/:/usr/bin/:/usr/sbin/:$PATH
export CPLUS_INCLUDE_PATH=/usr/local/gcc-5.5.0/include/c++/5.5.0/
export LD_LIBRARY_PATH=/usr/local/gcc-5.5.0/lib/
cd ..
```

> Not: Adımın `gmake` tamamlanması birkaç dakika sürecektir. Bu normal bir davranıştır.

6. cmake kütüphanesini kurun:

```
curl -sL http://packages.wazuh.com/utils/cmake/cmake-3.18.3.tar.gz | gtar xz
cd cmake-3.18.3
./bootstrap
gmake && gmake install
cd .. && rm -rf cmake-3.18.3
ln -sf /usr/local/bin/cmake /usr/bin/cmake
```

2. En son sürümü indirin.

```
git clone -b v4.9.2 https://github.com/wazuh/wazuh.git
```

> Not: Eğer Open SSL hatası nedeniyle dosyayı indiremiyorsanız, scp yardımcı programını kullanarak dizini kopyalamanız gerekir.

3. Betiği çalıştırın `install.sh`. Bu, Wazuh kaynaklarını kullanarak kurulum sürecinde size rehberlik edecek bir sihirbazı çalıştıracaktır:

```
cd wazuh*
./install.sh
```

4. Daha önce başka bir platform için derleme yaptıysanız, Makefile'ı kullanarak derlemeyi temizlemelisiniz `src`:

```
gmake -C src clean
gmake -C src clean-deps
```

> Not: Kurulum sırasında kullanıcılar kurulum yolunu belirleyebilir. Çalıştırın `./install.sh`ve dili seçin, kurulum modunu olarak ayarlayın `agent`, ardından kurulum yolunu (`Choose where to install Wazuh [/var/ossec]`) ayarlayın. Varsayılan kurulum yolu `/var/ossec`'dir . Yaygın olarak kullanılan özel bir yol `/opt` olabilir . Varsayılandan farklı bir yol seçerken, dizin zaten mevcutsa, yükleyici dizini silmeyi veya içine Wazuh'u yüklemeyi isteyecektir. Ayrıca gözetimsiz bir kurulum da çalıştırabilirsiniz .

1. Komut dosyası ne tür bir kurulum istediğinizi soracaktır. Wazuh aracısını kurmak için şunu `agent` yazın:

   Output

   ```
   1- What kind of installation do you want (manager, agent, local, hybrid, or help)? agent
   ```

Sonraki adımlar

Artık aracı yüklendiğine göre, bir sonraki adım aracı Wazuh sunucusuna kaydetmektir. Bu süreç hakkında daha fazla bilgi için lütfen Wazuh aracı kayıt bölümünü kontrol edin.

Kaldır

Wazuh aracısını kaldırmak için `WAZUH_HOME` geçerli kurulum yolunu ayarlayın:

```
WAZUH_HOME="/WAZUH/INSTALLATION/PATH"
```

Hizmeti durdurun:

```
service wazuh-agent stop 2> /dev/null
```

Daemon'u durdurun:

```
$WAZUH_HOME/bin/wazuh-control stop 2> /dev/null
```

Kurulum klasörünü ve tüm içeriğini kaldırın:

```
rm -rf $WAZUH_HOME
```

Hizmeti silin:

```
find /etc/{init.d,rc*.d} -name "*wazuh*" | xargs rm -f
```

Wazuh kullanıcı ve grubunu kaldır:

```
userdel wazuh 2> /dev/null
groupdel wazuh 2> /dev/null
```