

Procesi in prikriti procesi vmesnika API, ki jih uporablja Nova.

PROCES	OPIS
nova-api-service	Sprejema zahteve s strani končnih uporabnikov preko klicev API in se na njih odziva. Izvaja tudi večino orkestracij kih operacij (upravlja z instancami, omrežjem in slikami). Za to storitev Nova podpira tudi druge APIj-je kot so Compute API in Amazonov EC2 API.
nova-api-metadata service	Sprejema metapodatke instance preko lastnega ali pa EC2 API-ja. Metapodatki instance so IP naslov instance, ID instance, tip instance, ime gostitelja in druge. API se uporablja v multi-host načinu z nameščeno storitvijo nova-network.
nova-compute-service	Prikriti proces, ki zažene in konča instanco preko APIja hipervizorja. Prvo sprejme operacijo nad instanco iz sporočilne vrste, nato izvede niz sistemskih ukazov (primer zagon KVM instance) in osveži stanje v bazi. Podpira širok nabor hipervizorjev.
nova-schedule	Sprejema zahteve iz vrste in glede na trenutno porazdelitev obremenitve na strežnikih, ki se izvajajo računske storitve in tako določi na katerem računskem strežniku se bo navidezni stroj izvajal.
nova-conductor module	Vmesnik med procesom nova-compute in podatkovno bazo, ki preprečuje storitvi nova-compute neposredni dostop do baze, s tem omogoča horizontalno skaliranje, namesti se ga na računski strežnik.

nova-network	Prikriti proces, ki je podoben procesu nova-compute, sprejema opravila iz vrste in izvaja manipulacijo omrežja, kot je nastavljanje mostov v omrežjih ali spreminja pravila v tabelah IP.
nova-dhcpbridge	Sledi naslovom IP, ki se dodeljujejo navideznim strojem in jih shrani v bazo, tako zagotavlja da je vedno ista instanca na istemu naslovu.
nova-consoleauth daemon	Prikriti proces, ki avtenticira žetone uporabnikov, ki jih uporabljajo naslednja dva procesa.
nova-novncproxy daemon	Prikriti proces, ki omogoča povezavo VNC preko posrednega strežnika (ang. proxy) za dostop do navidezni strojev, ki se trenutno izvajajo.
nova- xvpnvncproxy	Posredni strežniški prikriti proces, ki dostopa do navideznih strojev, ki se trenutno izvajajo.
nova-cert daemon	Prikriti proces, ki služi za upravljanje s certifikati x509.
nova client	Omogoča uporabnikom delo preko ukazne vrstice kot končni uporabnik ali pa kot administrator.
nova-manage client	Omogoča dostop administratorjem do oblaka, ki imajo veliko pravic in avtorizacije.
nova-object store	Prikriti proces, ki omogoča komunikacijo med amaznovim vmesnikom S3 in vmesnikom za slikovno hrambo.

euca2ools client	Proces, ki prevaja iz ukazne vrstice amazonovega EC2 API-ja v ukaze za delo z oblakom.
------------------	--

Načini avtentikacije in avtorizacije komponente Keyston.

NAČINI AVTENTIKACIJE IN AVTORIZACIJE	OPIS
Uporabnik (<i>ang. user</i>)	Digitalna prezentacija osebe, sistema ali storitve, ki uporablja storitev oblaka. S pomočjo avtentikacije se prepoznajo uporabniki in njihove nadaljnje zahteve. Administrator dodeli dostopne pravice uporabniku, ki določajo kaj lahko uporabnik uporablja, katere operacije izvaja in količino virov, ki mu je na voljo. V kolikor uporabnik želi dostopati do virov, mora biti dodan tenantu.
Prijavni podatki (<i>ang. credentials</i>)	Poverilnice so podatki, ki pripadajo točno določenemu uporabniku s katerimi se avtenticirajo in ne smejo biti dostopni tretjim osebam. Primer: uporabniško ime in geslo, ujemanje uporabniškega imena in ključa APIja in žetoni vezani na uporabnike.
Avtentikacija (<i>ang. authentication</i>)	Je potrjevanje identitete uporabnika in njegovih zahtev. V kolikor se potrdi identiteta uporabnika, potem se njegove

	<p>zahteve sprejeme in izvede. Primer podatkov so prijavni podatki.</p>
<p>Žeton za dostop (ang. token)</p>	<p>Žeton vsebuje zapis o varnostnih elementih storitve, procesa, sistem in ima določeno do katerih virov lahko dostopa. Po določenemu času poteče.</p>
<p>Stanovalec (ang. tenant)</p>	<p>Stanovalec določa količino virov za uporabnike, račune, organizacije in projekte.</p> <p>Viri so VLAN, fiksni in plavajoči naslovi IP, število navideznih naprav in njihove kapacitete, število navideznih instanc, število vCPE, slike instanc, ključe in uporabnike).</p>
<p>Storitev (ang. service)</p>	<p>Storitev OpenStacka določajo več končnih točk preko katerih uporabnik lahko dostopa do posameznih komponent in jih uporablja.</p>
<p>Dostopna točka (ang. endpoint)</p>	<p>Dostopna točka ali končna točka v omrežju je računalnik ali konzola v omrežju, ki ima dostop do storitev oblaka. Določa jo URL naslov storitev.</p>
<p>Vloga (ang. role)</p>	<p>Administrator vlogo dodeli uporabniku in tako določi kateri uporabnik ima dostopne pravice do podatkov, storitev in katere operacije sme izvajati.</p>

Procesi komponente Glance.

PROCES	OPIS
glance-api	Sprejema ukaze API za odkrivanje, poizvedovanje in shranjevanje slik.
glance-registry	Shranjuje, poizveduje in procesira meta podatke o slikah, kot so: velikost slike, tip slike , ID slike in druge.

Procesi storitve Neutron.

PROCES	OPIS
neutron-server	Sprejema zahteve in jih pošilja vtičem (<i>ang. plug in</i>) znotraj omrežja.
omrežni vtičniki in agenti	Omogočajo in onemogočajo porte, ustvarjajo omrežja in podomrežja, izvaja naslavljanje IP. Agenti so L3 (<i>ang. Layer 3</i>), DHCP in drugi agenti.
sporočilne vrste	Prenašajo sporočila med strežnikom neutron in agenti. Shranjujejo stanje omrežij v podatkovni bazo.

Procesi komponente Cinder.

PROCES	OPIS
cinder-api	Sprejema zahteve APIjev in jih posreduje cinder-volume za nadaljnje procesiranje.

cinder-volume	<p>Shranjuje in bere stanje navideznih naprav v bazi. Sodeluje z ostalimi procesi (cinder-scheduler) preko sporočilne vrste.</p> <p>Na voljo ima veliko gonilnikov za shranjevanje podatkov pri ponudnikih tovrstnih storitev.</p>
cinder-scheduler	<p>Prikriti proces, ki strežnik na katerem je optimalno prostora za tvorjenje nove virtualne naprave. Prav tako uporablja sporočilne vrste z namenom posredovanja sporočil med procesi Cinderja ,shranjuje stanja navideznih naprav v bazo in Novi omogoča navidezne naprave za instance.</p>
sporočilne vrste	<p>V sporočilno vrsto se postavijo informacije, ki so potrebne med procesi Cinderja.</p>

Procesi komponente Swift.

PROCES	OPIS
swift-proxy-server (account server)	Posredniški strežnik za račune, ki so definirani s strani storitve za shranjevanje objektov. Sprejema ukaze swiftovega API-ja in HTTP zahteve za nalaganje slik, spreminja metapodatke in izdeluje vsebnike.
swift-account-server (container server)	Upravlja z računi, ki jih definira Swift.
swift-container-server (object server)	Strežnik, ki skrbi za upravljanje in mapiranje vsebine vsebnika ter map z objekti znotraj Swifta.
swift-object-server	Upravlja z dejanskimi objekti, kot so datoteke na shranjevalnih vozliščih (<i>ang. storage node</i>) oziroma strežnikih.
periodični procesi	Skrbijo za odstranjevanje podatkov na velikih podatkovnih shrambah (<i>ang. data store</i>). Na ta način zagotavlja konsistentnost in razpoložljivost gruče strežnikov. Ostali periodični procesi izvajajo revizije, posodobitve in ponavljanje opravila, dokler opravila niso zaključena.
vmesnik WSGI	Opravlja avtentikacijo za dostop do Swifta. Ta vmesnik je po navadi Keystone.

Procesi komponente Ceilometer.

PROCES	OPIS
ceilometer-api	Zbira podatki o statistiki uporabe vseh računskih strežnikov za naplačevalne sisteme in orodja za analizo podatkov.
ceilometer-agent-central	Zbira statistiko uporabe na kontrolnem vozlišču (ni povezano z instancami in računskem strežniku)
ceilometer-agent-compute	Od hipervizorjev pridobiva podatke o njegovih delovanju.
Ceilometer-agent-notification	Sprejema sporočila AMQP od ostalih komponent platforme.
ceilometer-collector	Zbira sporočila AMQP od agentov in jih dostavlja procesu za beleženje podatkov.
ceilometer-alarm-notifier	Zbira statistiko z več kontrolnih vozlišč in omogoča nastavitve sporočil o prekomerni uporabi storitev, doseženih pragovih prenosa podatkov in uporabe instanc.
podatkovna shramba	Je sposoben upravljati z več sočasnimi branji in pisanji, ki jih prejema od storitev telemetrije.

Procesi komponente Heat.

PROCES	OPIS
heat CLI	Komunicira s procesom heat-api in omogoča zaganjanje AWS CloudFormation API-jev preko katerih pridobi predlogo infrastrukture.
heat-api component	Proces REST API sprocesira zahteve namenskega OpenStackovega REST API-ja tako, da jih pošilja heat-engine preko RPC (<i>ang. remote procedure call</i>)
heat-api-cfn component	Proces je AWS Query API, ki je združljiv z AWS CloudFormation in zahtevo sprocesira do heat-engine preko RPC.
heat-engine	Zažene infrastrukturo iz predloge in se odziva na klice klijentovega API-ja.

Procesi komponente trove.

PROCES	OPIS
python-troveclient	CLI, ki komunicira z trove-api.
trove-api	Omogoča RESTful API preko katerega upravlja instance.
proces Trove-conductor	Se izvaja na strani gostitelja in sprejema podatke od instance.
Proces trove-taskmanager	Upravlja s sistemom, ki izvaja operacije nad instanco.
Proces trove-guestagent	Se izvaja na gostujoči instanci in izvaja operacije nad podatkovno bazo.