Provozní standardy

Seznam jednotlivých oblastí, kterými se je Poskytovatel povinen řídit:

A. Převzetí aplikací a služeb do OCIS infrastruktury
B. Podporované provozní standardy OCIS MV ČR pro správu a provoz aplikací
C. Podporované provozní technologické standardy OCIS MV ČR pro platformy Oracle Solaris Sparc, Linux CentOS a Debian

Obsah:

A. Převzetí aplikací a služeb do OCIS infrastruktury

1 Úvod

Standard popisuje postup, převzetí aplikací a služeb do infrastruktury Ministerstva vnitra, odboru centrálních informačních systémů (dále OCIS).

1.1 Účel a podmínky

Definovat proces, převzetí aplikací a služeb do správy specialistů OCIS a Národní agentury pro komunikační a informační technologie, s.p. (dále NAKIT).

Proces se uplatňuje na aplikace a služby, které bude OCIS provozovat nebo podporovat prostřednictvím specialistů OCIS/NAKIT, nebo 3. strany.

Hlavním cílem procesu je:

- Testováním ověřit, že aplikace nebo služba přebíraná do provozní podpory:
  - Splňuje definované požadavky;
  - Odpovídá dokumentovanému návrhu;
  - Splňuje akceptační kritéria.
- Poskytnout výsledky testování a ověření splnění akceptačních kritérií.
- Rozhodnout o řešení situace, kdy akceptační kritéria nejsou splněna.
- Zajistit veškeré zdroje pro následný provoz aplikace nebo služby:
  - Interní lidské zdroje;
  - Technické zdroje;
  - Servisní krytí;
  - Licence pro veškerý použitý SW.
- Zavést jednotlivé provozní procesy
- Uskutečnit uvedení do provozu a zahájit provoz.

1.2 Rozsah působnosti

Postup je závazný pro zaměstnance organizačních jednotek OCIS, NAKIT a zaměstnance třetích stran ve smluvním vztahu s OCIS, kteří zajišťující vývoj, testování, implementaci, správu, provoz a podporu aplikací.

2 Popis postupu

2.1 Povinné vstupy

- Dokumentace skutečného provedení
  Dokumentace navazuje na design a obsahuje všechny detaily o provedené implementaci především volbu identifikátorů a parametrů, které byly v průběhu implementace nastaveny.
- Konfigurační položky
  Seznam konfiguračních položek na základě skutečného provedení pro úvodní vložení do konfigurační databáze Glpi.
- Seznam smluv o podpoře (**maintenance list**) pro každou komponentu řešení získanou od třetí strany (HW/SW):
  - Charakter podpory;
  - Provozní doba podpory, časové lhůty a úroveň SLA;
  - S podporou související smlouvy (určující dodavatele a nákupní podmínky).
- Pro každou SW komponentu řešení:
  - Způsob licencování;
  - Informaci o počtu licencí;
  - Informaci době platnosti licencí;
  - S licencemi související smlouvy (určující dodavatele a nákupní podmínky).
- Testovací scénáře
  Dokumentace obsahuje jednotlivé kroky testů a jejich očekávané výsledky.
- Postup pro zřizování, změny a rušení aplikace, služby.

2.2 Akceptační testování

Cílem akceptačních testů je ověřit, že aplikace nebo služba přebíraná do provozní podpory:

- Splňuje definované požadavky;
- Odpovídá dokumentovanému návrhu;
- Splňuje akceptační kritéria.

Výsledky testů jsou zkompletovány do protokolu o testech:

- **APL_XXX_VER_YYY_evidenční_formulář**

Protokol je a uložen do knihy serverů (Glpi).

V případě odmítnutí akceptace musí vedoucí oddělení a specialisté OCIS/NAKIT vyřešit všechny neshody, které blokují akceptaci.
Důvodem k odmítnutí akceptace může být kromě nesplnění akceptačních kritérií nedodání povinných vstupů.

3 Související dokumentace

3.1 Řídicí dokumenty

- Dokumentace GLPI
- Pravidla vzdáleného přístupu
- Provozní standardy Unix/Linux
- Provozní standardy NT
- Provozní standardy pro správu a provoz aplikací
- Standard informační bezpečnosti pro platformy Oracle Solaris Sparc
- Standard informační bezpečnosti pro platformy Linux CentOS a Debian
- Standard informační bezpečnosti pro platformy Windows
- Standard monitoringu Zabbix
- Standard dohledu Graylog
- APL_XXX_VER_YYY_evidenční_formulář

B. Podporované provozní standardy OCIS MV ČR pro správu a provoz aplikací

1 Úvod

Standardizace je jednou z klíčových podmínek pro dosažení garantované kvality a udržitelné ceny provozu infrastruktury Ministerstva vnitra, odboru centrálních informačních systémů (dále OCIS).

1.1 Obsah a účel dokumentu

Definovat aktuálně podporované aplikační standardy – schválené parametry, standardizovaná nastavení aplikací a seznam požadavků pro testování, instalaci a provoz aplikací ve správě specialistů OCIS a Národní agentury pro komunikační a informační technologie, s.p. (dále NAKIT), které jsou podmínkou pro zvládnutí rychlého a úspěšného nasazení a hlavně jejich následného garantovaného provozování a podpory.

1.2 Rozsah působnosti

Postup je závazný pro zaměstnance organizačních jednotek OCIS, NAKIT a zaměstnance třetích stran ve smluvním vztahu s OCIS, kteří zajišťující vývoj, testování, implementaci, správu, provoz a podporu aplikací.

1.3 Rozsah tohoto standardu

Předkládaný aplikační standard se vztahuje na aktuálně spravované aplikace a na všechny nové aplikace (vč. změn, nových verzí a aktualizací stávajících aplikací) akceptované do provozu OCIS.

1.4 Předpoklady pro správné fungování aplikačního standardu

- Prostředí aplikace musí být v souladu se souvisejícími standardy infrastruktury – DC / HW / OS / sítě / clustery / storage / DB / AS / WEB, koncová zařízení
- Prostředí aplikace musí být v souladu se standardy bezpečnosti
- Prostředí aplikace musí odpovídat podmínkám nasazení a akceptace aplikací do provozu

Seznam souvisejících dokumentů zmiňovaných standardů a požadavků je uveden v kapitole 3.

1.5 Možné výjimky ze standardů a jejich řešení

Tato kapitola obsahuje popis chování útvarů provozu OCIS Infrastruktura v případě možných výjimek a určuje, jakým způsobem se řeší konflikty ve standardech, kdo má finální odpovědnost a jaké jsou s tím spojené podmínky.

- V případě kolizních požadavků, které odporují standardům, nebo jinak ohrožují provoz aplikací, může instalaci nové, nebo změnu již instalované aplikace, zamítnout provozovatel, resp administrátor aplikace a rozhodnutí nahlásí vedoucímu oddělení.
- Akceptace aplikace do provozu je možná pouze za předpokladu již dokončené akceptace související infrastruktury. Aplikace nasazované na infrastrukturu, která neprošla akceptací specialistů OCIS/NAKIT, nejsou standardně podporovány.
- Veškeré nestandardní požadavky na provoz aplikací je nutné předem řešit s provozovatelem příslušné aplikace.
- Aplikace a/nebo její součásti (DB, skripty, moduly, atd), jejichž funkčnost či vlastnosti odporují standardům, nebo jiným způsobem nevyhovují (např. nedostatečná dokumentace, neodpovídá bezpečnostním standardům apod.), OCIS Infrastruktura, jako provozovatel odmítá k převzetí do provozu. V případech, kdy je převzetí takovýchto nestandardních a běžným postupem neakceptovatelných aplikací a jejich součástí do provozu ve zvýšeném zájmu uživatele, může OCIS souhlasit s podmínečnou akceptací do provozu za zvláštních podmínek:
- Aplikace je zařazena do zvláštní kategorie (X = Speciální).
- Pro aplikace v kategorii „X“ NEJSOU poskytovány služby ve standardní úrovni a rozsahu (nemusí být dostupné služby pro dohled, zálohování, apod.).
- Kritičnost aplikace a s tím související úrovně služeb (dostupnost, rychlost odezvy na incident, apod.) stanovuje OCIS Infrastruktura na základě vlastní analýzy. Aplikace v kategorii X jsou obvykle provozovány na základě parametrů kategorie Standard, čemuž odpovídá úroveň poskytovaných služeb. Podporu aplikací v kategorii X řeší OCIS Infrastruktura na bázi „Best Effort“, tzn., že kvalita služeb SLA není garantovaná a tedy žádným způsobem vymahatelná.
- Do kategorie X spadají i aplikace, které ještě nebyly standardním způsobem předány do provozu, nicméně je žadatelem požádáno o poskytnutí provozní podpory např. ještě v době vývoje (typicky instalace nových řešení).

Nestandardní aplikační řešení zařazené v kategorii X generují zvýšené náklady, které jsou spojené s jejich specifickou správou. OCIS Infrastruktura má právo v rámci akceptace nestandardní aplikace vyčíslit tyto dodatečné náklady (navýšení FTE) a požadovat je po žadateli.

2 Popis aplikačního standardu

2.1 Označení aplikace

Skládá se z názvu a pokud je to možné, tak i verzí.

2.1.1 Název aplikace

Název aplikace musí splňovat:

- Unikátnost v rámci Glpi (Komplexní nástroj na management a správu IT prostředků)
- Nesmí obsahovat rezervované ani speciální znaky jako je mezera, speciální znaky využívané operačním systémem (např. "/" a "\"), v příkazech SQL (např. "%" či "?") znaky používané v zřetězení příkazů v UNIXu/Windows – např. "|", "<", ">" a pod.
- Musí to být celistvý jednoduchý řetězec znaků – velkých a malých písmen a číslic v délce maximálně 15 znaků

2.1.2 Verze

Verze je tvořena číselnou sekvencí v jednom z následujících formátů

[MAJOR_VERSION].[MINOR_VERSION].[REVISION_NUMBER].[BUILD_

NUMBER] kde:

- [MAJOR_VERSION] označuje verzi SW, která obsahuje rozsáhlé funkční změny (např. včetně změny datového modelu), popř. dopředu nebo dozadu je nekompatibilní s předchozími/budoucími verzemi – vyžaduje specifické provedení upgradu
- [MINOR_VERSION] označuje verzi, která přidává nějaké funkční vlastnosti, případně mění GUI, ale je zpětně kompatibilní
- [REVISION_NUMBER] zpravidla označuje pouze drobné opravy funkčnosti, ale neznamená změnu funkčnosti aplikace
- [BUILD_NUMBER] zpravidla číslování ze strany dodavatele, popř. vývojáře SW – často se týká úrovně Service packů.

Pokud verzování není možné, doporučuje se akceptovat způsob značení z pohledu dodavatele SW a to i v případě, že jde o alfanumerické označení.

2.2 Kategorie aplikace

Kategorie aplikace je odvozena z relativní důležitosti – „kritičnosti" – aplikace pro fungování procesů ve společnosti. Ve vazbě na to určuje požadavky na kontinuitu, dostupnost a další provozní parametry. V závislosti na stupni kritičnosti jsou stanoveny 3 kategorie:

MC Mission Critical – HA + DR řešení, provoz 7x24

C Critical – HA řešení, provoz 7x24

S Standard – bez HA řešení, provoz v pracovní době 5x8

Kritičnost aplikace musí být nastavena před zahájením rutinního provozu, nejpozději v procesu akceptace do provozu.

2.3 Požadavky na nasazení

Veškerý aplikační SW a jeho změny musí být do provozu nasazovány prostřednictvím SW dávek a pracovních instrukcí.

Pro instalaci musejí být vytvořeny instalační skripty a procedury, které musejí splňovat tyto podmínky:

1. Instalační procedury musí počítat s výskytem situace „rollback“, tzn. že nově nasazenou změnu je nutné odrolovat zpět a namísto ní použít předchozí verzi aplikace.
2. Instalační programy a skripty musí během procesu instalace jednoznačně informovat administrátora provádějícího instalaci dle postupu, nejlépe návratovým kódem a informační zprávou, o výsledku procesu: úspěch / neúspěch.
3. Instalační programy a skripty musí z procesu instalace vytvářet jednoznačný log/protokol, ze kterého je možné i zpětně verifikovat průběh a výsledek celého procesu nasazení změn.
4. Instalační programy a skripty musí minimalizovat nutnost používání práv na superuživatele (root, Administrátor).

- V případech, kdy je použití práv nutností, musí instalační procedura obsahovat samostatný bod (skript), k jehož spuštění dojde za asistence administrátora příslušného OS.
- Instalační skripty nesmí automaticky vytvářet uživatele v OS/DB. Požadavek aplikace na vytvoření uživatele se musí vypořádat jako součást předpokladů pro instalaci (viz Provozní standardy).
- Instalační skripty nesmí zakládat adresáře/soubory mimo adresářovou strukturu dedikovanou pro aplikaci (popis požadavků na adresářovou strukturu je součástí předpokladů pro dokumentaci).

5. Musí být dodán instalační manuál.
6. Provozní dokumentace musí obsahovat i případné specifické požadavky na profylaxe a jiné proaktivní kroky zabezpečující spolehlivost a dostupnost aplikací v souladu s příslušným SLA.
7. V rámci akceptace musí dojít i k nastavení podpory v procesech řízení incidentů a správa problémů.

Nejpozději při akceptaci musí být známo obsazení rolí garant aplikace.

Poznámka: Totéž platí pro instalaci změn.

2.4 Správa a provoz aplikace

Na provozu aplikace se podílí:

- správa aplikací
- správa DB
- správa sítí
- správa koncových zařízení
- správa datových center
- správa middleware
- správa infrastruktury

Pro zachování správného fungování aplikace musejí být komplexně respektovány příslušné standardy ve všech těchto částech provozu.

2.5 Plánování a provádění servisních odstávek

Odstávková časová okna pro servisní potřeby, release a případně další potřeby musejí být specifikována v rámci SLA, dle kritičnosti dané aplikace.

Poznámka: Odstávky TEST, DEV a STANDARD prostředí se provádějí v pracovní době 8:00 – 16:00 po dohodě s příslušným garantem aplikace.

2.6 Spouštění pravidelných (dávkových) úloh

- Pro možnost spouštění pravidelných batchových úloh či jiných periodických operací (extrakty, zálohování, spouštění jobů, SQL skriptů apod.) jsou povoleny následující způsoby:
    - vlastní plánovač – aplikace implementuje v rámci svého kódu plánovač (např. Quartz pro J2EE aplikace), který zajišťuje spouštění periodických akcí podle definovaného plánu
    - plánovač v OS (cron (UX), Task Scheduler (Windows), apod.) – aplikace využívá plánovač v OS;
    - manuální spouštění – ve výjimečných situacích (např. při nutnosti navázat batch úlohu na jinou manuální činnost), je akceptovatelné spouštění úlohy proškoleným operátorem;
- Podklady pro nastavení plánovače, včetně případných manuálních procedur, musí být součástí dokumentace (termín a frekvence spouštění, obvyklá délka běhu, atd).

- Sledování výsledku (i běhu) batchové úlohy musí být možné zahrnout do nepřetržitého dohledu v souladu se standardy viz. Související dokumentace v kap.3 (nejlépe sledování zpráv v logu).

2.7 Logování a jeho pravidla

- Pro korektní správu aplikací se požaduje provádět logování (používání žurnálových souborů). Většina standardních komerčních aplikací tuto funkcionalitu nativně poskytuje. Při logování specificky na úrovni aplikací se vyžaduje:

    - logovat **vždy** kritické chyby aplikace (pády aplikačních serverů či DB, fatal errors)
    - volitelně logovat závažné chyby aplikací (např. neprovedení DB transakce, chyby Informix apod.)
    - logovat závažné chyby podpůrných částí aplikací (DB, sítí, webových serverů) apod.
    - logovat vždy, pokud je to technicky možné s ohledem na bezpečností standardy přístupy uživatelů do aplikace (minimálně logovat neúspěšné pokusy o přihlášení)
    - ohledem na riziko potenciálního přetečení diskového prostoru, všude kde to charakter aplikace umožňuje provádět cyklické přepisování logů, vždy však ve spolupráci s archivací dat, aby nedošlo ke ztrátě informací

2.7.1 Obsah logu

Aplikace musí logovat všechny důležité stavy v různých oblastech provozu a to zejména:

- Celková funkčnost / dostupnost / změny stavu
    - Start / stop aplikace / procesu / funkce
    - Re-load konfigurace, apod.
    - Keep-alive zprávy (indikující, že aplikace běží, i když by jinak nepsala do logu)
- Bezpečnost
    - Autentizace uživatele (success / failure)
    - Přístup na citlivá data (přístup do DB, apod.)
    - Chyba validace vstupních parametrů (pokus o podvržení dat)
- Trasování zpracování (debug)
    - Přijetí požadavku => parsování, zpracování, commit/rollback => odpověď
    - Vytvoření DB spojení, spuštění SQL příkazu, apod.
    - Detailní trace zpracovávaných dat (parametry volání, apod.) pro účely ladění a testů
- Chyby a neočekávané výjimky
    - Nedostupné související systémy nebo jejich pomalá odezva (DB, web-server, MW konektor, apod.)
    - Nedostatečné místo na disku
    - Neodchycená výjimka, apod.
    - Vypršení timeoutu volané funkce
- Statistické informace
    - Přístupy na určitou funkci, stránku, apod. (např. statistiky web serverů)

2.7.2 Struktura logů

- Logy musí mít popsanou strukturu (pevný počet polí, shodné oddělovače záznamů).
- Musejí být konstruovány tak, aby šly parsovat běžnými nástroji nebo musí být logy konfigurovatelné (např. prostřednictvím frameworku log4j, apod.).
- Formát logu musí podporovat snadné vyhledávání dle klíčových slov. Klíčová slova musí jednoznačně identifikovat důležitost zprávy (severity levels)
- Jednotlivé záznamy v logu musí kromě zprávy obsahovat časové razítko (formátované dle TZ a LOCALE serveru).
- Jednotlivé záznamy v logu by dále měly obsahovat dodatečné informace sloužící ke snadnějšímu zařazení zprávy (thread/proces generující zprávu, volaná metoda/funkce, apod.).
- Součástí dokumentace k aplikaci musí být popis všech významných logovaných zpráv spolu s popisem jejich významu. Seznam chybových zpráv musí obsahovat i doporučené řešení příslušného chybového stavu.

- V konfiguraci loggeru se požaduje možnost zvýšit/snížit úroveň důležitosti logovaných zpráv. Součástí dokumentace k aplikacím musí být doporučené implicitní nastavení logování pro běžný provoz, jako i nastavení logování pro případ řešení chyb a problémů (trasování, debug).

2.7.3 Rotování a archivace logů

- Musí být specifikovány předpokládané nároky na diskové kapacity pro logy (řeší se v rámci celkových požadovaných diskových kapacit pro aplikaci).
- Aplikace musí implementovat pravidelnou rotaci logů, která by měla být konfigurovatelná. K uzavření logu by mělo dojít po dosažení definované velikosti, nebo při dosažení časového okamžiku (denně, týdně).
- Pokud aplikace neimplementuje pravidelnou rotaci logů, musí být v dokumentaci uvedeny požadavky na pravidelnou rotaci (a navržený způsob rotace).
- Dokumentace logů musí obsahovat požadavky na případnou retenci logů. Případné požadavky na jejich delší uschování, např. z důvodu legislativních požadavků, je nutné řešit v souladu se zásadami zálohování resp. dlouhodobé archivace dat.

2.8 Backup / Recovery / Archivace

2.8.1 Zálohování, obnova a archivace

Postup zálohování/obnovy, případně dlouhodobé archivace dat aplikace musí být uvedeno dokumentaci.

Pokud aplikace vyžaduje specifické kroky, např. vykonání určité sekvence příkazů jako součást zálohy, musí být součástí dodávky aplikace skript realizující potřebné příkazy.

Aplikace s požadavkem na nepřetržitý provoz musí podporovat vytvoření konzistentní on-line zálohy dat bez odstávky provozu aplikace.

2.8.2 Disaster Recovery (kontinuita)

Pokud aplikace vyžaduje specifické kroky (např. vykonání určité sekvence příkazů jako součást převodu provozu), musí být součástí dodávky aplikace skript(y) realizující potřebné operace.

2.9 Požadavky na konfiguraci aplikace

2.9.1 Flexibilní konfigurace

Aplikace musí mít možnost konfigurovat (změnit dle potřeb provozu) důležité aplikační parametry:

- Umístění adresářů/souborů aplikace:
  - Umístění statických dat aplikace (cesta na adresář(e) s binárními a konfiguračními soubory),
  - Umístění dynamických dat aplikace (cesta na adresář(e))
  - Změna umístění vstupních/výstupních či jiných souborů,
  - Změna vlastníka a práv souborů/procesů aplikace
- Parametry DB:
  - Změna připojovacího řetězce k DB (změna jména DB instance)
  - Změna uživatele používaného pro přístup k DB
  - Změna parametrů DB:
  - Umístění schématu v DB (např. jiný tabulkový prostor)
  - Parametry (např. pojmenování) tabulkových prostorů
- Přístupové parametry:
  - Aplikační servisní (technologické) účty, účty uživatelů/administrátorů aplikace, a jejich hesla
- Komunikační parametry:
  - Doménová jména (FQDN), čísla TCP/UDP portů
  - Aplikace musí podporovat resolving pomocí FQDN, protože používání krátkých jmen a IP adres je zakázáno!

Konfigurace aplikačních parametrů musí být dostupná aplikačnímu administrátorovi pomocí konfiguračních souborů (běžné formáty: txt, xml, apod.), konfiguračních položek v DB, alternativně pomocí GUI.

Změny základních parametrů aplikací, které vyžadují rekompilaci kódu aplikace či obdobné postupy, jsou nestandardní a je nutné k nim vyžádat výjimku.

Aplikace musí umožňovat oddělení statických dat (binární soubory/knihovny, konf. soubory) a dynamických dat (datové soubory, apod.). Proces nasazení aplikací implicitně předpokládá, že všechny aplikace mají data různé povahy oddělená a pro jiný stav je tedy nutno žádat výjimku.

Popis a návod ke změnám veškerých konfiguračních parametrů musí být součástí dokumentace.

2.9.2 Aplikace JAVA/J2EE

- Všechny aplikace typu J2EE musí používat pro svůj běh aplikační servery
    - Provoz JAVA aplikace mimo aplikační server je možný pouze za předpokladu schválení adekvátně odůvodněné výjimky.
- JAVA aplikace (či její aplikační server) NESMÍ využívat implicitní JRE/JDK instalovanou v OS pro účely správy a provozu OS, které jsou umístěné obvykle v následujících cestách:
- UNIX: /opt/java* , /opt/jdk*, /usr/java*, /usr/j2se*, /usr/jdk*
- WINDOWS: %SYSTEMROOT%/system32, ?:\Program Files\Java*.*
- Aplikace využívající JRE/JDK musí obsahovat a používat vlastní JRE/JDK build, který je umístěný v adresářové struktuře určené pro binární soubory aplikace.
- V konfiguraci aplikace platí povinnost změnit generické jméno procesu „java“ na jiné (mnemotechnické pojmenování procesu, které lépe reprezentuje běžící proces)
- Logování GC (Garbage Control[1])– pro každou běžící JVM konfigurovat podrobné logování statistik GC (volby: -verbose:gc , nebo -Xverbosegc (HP only) apod.)
- Konfigurace paměti pro JVM – se požaduje pro každou běžící JVM s vyššími nároky na výkon konfigurovat:
- min/max paměti heapu na stejnou statickou hodnotu (např. –Xmx1024m –Xms1024m)
- Použití velkých paměťových stránek
- Pro více informací viz dokumenty na téma JAVA Perf. Tuning.
- Parametry JVM nesmí implicitně obsahovat vytvoření heapdumpu (např. jako důsledek stavu Out of memory error), aby nedošlo v případě jeho výskytu k neočekávanému zaplnění filesystému. Dočasná rekonfigurace aplikace za účelem vytvoření heapdumpu je však možná při eskalaci produkčního problému, který nelze simulovat jinak než v produkci. Změnu konfigurace JVM pak provádí IT Infrastruktura – správa aplikací v souladu s požadavkem/instrukcí zadavatele.

[1] Java používá termín Garbage Control ve smyslu alokování, resp. spíše nealokování dynamicky obsazované paměti. Souvisí to se způsobem dynamického alokování paměti (heapu) při vytváření objektů v Javě. V případě nekorektního a nebo úplně zapomenutého ošetření tzv. výjimek. V kódu může dojít k tomu, že se tato paměť neuvolňuje a časem dochází k přetečení paměti.

2.10 Programy a skripty

- Všechny aplikační programy, moduly a skripty musí spolehlivě ošetřit a konzistentně vracet návratové hodnoty:
    - 0 = úspěšný běh (OK)
    - >0 = neúspěšný běh (ERR)
- Součástí aplikace a všech jejích součástí (modulů, DB, ...) musí být spolehlivé a funkční skripty/programy, které slouží pro spuštění a zastavení aplikace. Požadované volby skriptů a jejich očekávaná standardní funkčnost jsou následující.
- start
- kontrola, zda aplikace běží;
- pokud neběží, skript aplikaci nastartuje a poté provede kontrolu, zda aplikace kompletně nastartovala
- OK = vypíše informativní zprávu a vrací kód 0
- ERR = vypíše varovnou zprávu a vrací kód >0

- pokud aplikace běží, vypíše pouze informativní zprávu a vrací kód 0
- stop
- kontrola, zda aplikace běží;
- pokud běží, skript aplikaci zastaví a poté provede kontrolu, zda aplikace kompletně zastavila
- OK = vypíše informativní zprávu a vrací kód 0
- ERR = vypíše varovnou zprávu a vrací kód >0
- pokud aplikace neběží, vypíše pouze informativní zprávu a vrací kód 0
- status
- kontrola, zda aplikace běží;
- pokud běží, vypíše informativní zprávu a vrací kód 0
- pokud aplikace neběží, vypíše varovnou zprávu a vrací kód > 0
- Startovací skripty nemusí vždy používat exaktně parametry start|stop|status, ale musí poskytovat odpovídající funkčnost, tj. musí umožnit spolehlivé zastavení a spuštění aplikace.
- Pokud nefunguje spolehlivě start/stop procedura, která zastaví/spustí aplikaci na jeden pokus, start/stop skripty pak musí obsahovat ošetření tohoto stavu. Počet pokusů o start/stop aplikace musí být konfigurovatelný administrátorem aplikace. Základní funkce skriptů, schopnost spolehlivě nastartovat/zastavit aplikaci, však musí zůstat zachována, jak je popsáno výše.
- Pokud aplikace nebo její součásti pro svůj start bezpodmínečně vyžadují dostupnost související komponenty (např. DB), startovací skripty musí provést kontrolu, zda jsou tyto komponenty dostupné ještě před pokusem o start aplikace. Cílem tohoto požadavku je zamezení zbytečného pokusu o start aplikace, jenž nemá dostupné kritické závislosti v době svého startu, a s tím spojenou časovou prodlevu, která může negativně ovlivnit požadovanou dostupnost. Pokud startovací skript tuto kontrolu závislostí provést neumí (např. z důvodu nedostatečných přístup. práv), musí na druhou stranu spolehlivě a bez prodlení informovat o nepovedeném startu aplikace.
- Startovací skripty musí být možné integrovat do inicializačních procedur OS (init, svc atd. (UNIX), service (Windows).
  Skripty proto nesmí vyžadovat zadání vstupních parametrů ze standardního vstupu (STDIN), apod.
- Skripty a programy nesmí pro svůj běh vyžadovat zadání hesla. Potřebné přístupové parametry (např. hesla, resp. jejich jednosměrný hash) musí být umístěny v konfiguračních souborech.
- Pokud uvedený postup není možný, tzn. heslo nejde zadat ve formě bezpečného hash, nebo není automatický start bez hesla žádoucí (ochrana privátního klíče, apod.), musí být zavedeny specifické manuální procedury a aplikace je označena značkou „vyžaduje zvláštní přístup“. Tuto situaci je nutno oznámit a zdokumentovat.

2.11 Požadavek vysoké dostupnosti (High Availability)

Obecná pravidla jsou tato:

- u aplikace provozované v režimu HA musí způsob zajištění DR odpovídat požadavkům na dostupnost aplikace (kritičnost).
- Aplikace, které mají SPOF na úrovni některé důležité komponenty či konektoru, nelze provozovat na úrovni standardní SLA poskytované pro kategorie MC a C.
- Aplikace v kategorii C a MC musí přímo podporovat běh v rámci HA clusteru nebo load balanceru
- Podpora a nasazení clusterů pro aplikace v kategorii S není mandatorní, ale je doporučená.
- Clusterová řešení a řešení pro load-balancing musí odpovídat seznamu podporovaných platforem.
- Aplikace instalované v HA clusterech (failover cluster) musí používat příslušné FQDN (DNS) pro přístup k aplikaci běžící v clusteru po IP síti (virtuální doménové jméno odpovídající příslušně virtuální IP adrese).
- Aplikace instalované v HA clusterech (distribuovaný cluster – load-balancing) musí používat příslušné FQDN (DNS) pro přístup k aplikaci běžící v clusteru po IP síti (IP alias příslušný dané službě, výjimečně lze povolit i doménové jméno serveru).

- Aplikace instalované v HA clusterech (failover cluster) musí mít možnost umístit dynamická data (DB, fronty, apod.) a alternativně i statická data na sdílenou storage v SAN.
- Aplikace v kategorii MC musí mít v době akceptace do provozu funkční a otestované DR procedury (uživatel zajišťuje podklady pro DR procedury).
- Komponenty vícevrstvé architektury (AS-DB, WEB-AS-DB), musí vzájemně podporovat automatické obnovení spojení se souvisejícími vrstvami v případě krátkodobé či delší nedostupnosti jedné vrstvy.
    - Např. při výpadku databáze (failover instance v HA clusteru, restart DB z jiného důvodu, apod.) provede aplikační server automatickou obnovu připojení (např. refresh JDBC poolu) v definovaném čase bez nutnosti restartu aplikačního serveru či jiných komponent.
    - Požadavek je mandatorní pro aplikace s důležitostí MC a C, doporučený pro aplikace v úrovni S.
- Integrační rozhraní různých komponent (DB linky, MW konektory, apod.) musí podporovat automatické obnovení přerušeného spojení se souvisejícími systémy/DB/aplikacemi v případě krátkodobé či delší nedostupnosti návazného systému.
    - Automatické obnovení spojení je mandatorní pro interface aplikací v kategorii MC a C, které jsou klíčové pro provoz těchto aplikací. Jinými slovy, nefunkčnost příslušného interface způsobí částečnou nebo celkovou nedostupnost hlavních funkcí aplikace nebo jinak ovlivní provozní parametry aplikace.

2.12 Správa uživatelských přístupů k aplikacím

Obecné principy jsou:

- Aplikace musí běžet pod definovaným uživatelem v OS, bez nutnosti alokace vyšších privilegií (root (Unix/Linux),
  Administrátor (Windows)); Výjimku z tohoto pravidla mají vybrané procesy, které z titulu svojí funkčnosti vyžadují elevaci práv (např. Web server běžící na portu 80).
- Pokud jsou k provozu a správě aplikace nutné různé úrovně přístupových práv, aplikace musí implementovat oddělení rolí.
- Pro údržbu uživatelských účtů (přidávání uživatelů, změnu profilů, přístupových práv, apod.) musí být k dispozici nástroj na správu. Tento požadavek neplatí, pokud jsou přístupová práva řízena automaticky prostř. centrálního řešení.
- Dokumentace k aplikaci musí obsahovat postup změn v uživatelských účtech, profilech, atd.
- Jednotlivé (koncové) uživatele musí být možné přidávat za běhu aplikace. Servisní (technologické) účty se obvykle přidávají v době odstávky).
- Způsob správy vlastních uživatelských přístupů musí být popsán v administrátorské příručce.

Správa vlastních uživatelských přístupů se provádí podle zvolené metodiky pro jednotlivé aplikace (musí být popsáno v administrátorské příručce). Vlastní správa obnáší přidělování a odnímání uživatelských přístupů buďto na úrovni vlastní aplikace, nebo pod ní běžící databáze, příp. obojí.

2.13 Nastavení spojení na DB

Musí být uvedeno v dokumentaci k dané aplikaci.

2.14 Dohledy a monitorování

- Aplikace musí podporovat implementaci dohledu a monitoringu svého běhu v souladu se standardy.
    - Standard monitoringu Zabbix
    - Dohled dokumentace Graylog a napojení do DCeGOV.
- Podklady pro implementaci automatického dohledu musí být součástí podkladů předávaných k akceptaci do provozu.
- Dokumentované procedury pro dohled musí umožňovat kontrolu funkčnosti zpracování, tzn. nejen pouze kontrolu dostupnosti běžících procesů aplikace/DB, ale zejména kontrolu funkčního zpracování.
- Pro aplikace musí být k dispozici monitorovací skripty/programy spustitelné z příkazové řádky. Skripty nesmí vyžadovat zadání vstupních parametrů ze standardního vstupu (STDIN), apod.

- Aplikace může implementovat GUI, nejlépe ve formě webového rozhraní, které umožňuje stejně jako dohledové skripty (viz kapitolu Programy a skripty) základní a kompletní informaci o funkčnosti aplikace (běží / neběží).

2.15 Aktualizace a aplikace patchů (service packů)

U každé aplikace musí být uvedeno a vyplněno ve formuláři:

- Způsob plánování aktualizací a upgradů (release policy)
- Jak se provádí aplikace patchů/service packů a jaké jsou podmínky nasazení (délka odstávky, způsoby provedení atd)
- Jak, kdy a za jakých podmínek se provádí upgrade aplikace na vyšší verzi (jaké podmínky musí být splněny – např. výkonnostní problémy, bezpečnost apod.) včetně řešení závislostí na
    - aktualizaci MW (např. dohrání nových knihoven apod.)
    - aktualizaci DB (např. patche či upgrady na nové verze)
    - aktualizaci OS (patche či upgrady na nové verze)

Formulář:

- APL_XXX_VER_YYY_evidenční_formulář

2.16 Bezpečnost

Je nutné splnit tyto požadavky:

- Aplikace odpovídá základním požadavkům na bezpečnost
- Součástí musí být schéma zónového modelu.
- U každé aplikace musí být uvedena senzitivita dat, podle které se rozhodne, kam aplikace patří.
- Omezení přístupových práv k souborům aplikace a všem jejím součástem pouze na skupinu uživatelů, která má mít z titulu svojí role přístup k datům. Jinými slovy je zakázáno používání práv přístupu pro všechny:
    - UNIX/Linux: práva pro skupinu Other (world readable/writeable)
    - Windows: Přístup pro skupinu Everyone, Users, apod.

3 Související dokumentace

3.1 Řídicí dokumenty

- Dokumentace GLPI
- Pravidla vzdáleného přístupu
- Provozní standardy Unix/Linux
- Provozní standardy NT
- Standard informační bezpečnosti pro platformy Oracle Solaris Sparc
- Standard informační bezpečnosti pro platformy Linux CentOS a Debian
- Standard informační bezpečnosti pro platformy Windows
- Standard monitoringu Zabbix
- Standard dohledu Graylog
- APL_XXX_VER_YYY_evidenční_formulář

C. Podporované provozní technologické standardy OCIS MV ČR pro platformy Oracle Solaris Sparc, Linux CentOS a Debian

1 Úvod

Standardizace je jednou z klíčových podmínek pro dosažení garantované kvality a udržitelné ceny provozu infrastruktury Ministerstva vnitra, odboru centrálních informačních systémů (dále OCIS).

1.1 *Obsah a účel dokumentu*

Dokument obsahuje schválené nastavení HW/SW a seznam požadavků pro instalaci a provoz serverů s operačními systémy typu UNIX/Linux. Dokumentuje také vedle postupu implementace prostředí i další pravidla platná pro provoz různých typů prostředí ve správě správě specialistů OCIS a Národní agentury pro komunikační a informační technologie, s.p. (dále NAKIT).

Pro zvládnutí rychlé a úspěšné integrace systémů s OS typu UNIX/LINUX a jejich aplikací do OCIS infrastruktury je bezpodmínečně nutné dodržet všechny podmínky pro jejich uvedení do provozu. Podmínkami pro provoz se rozumí:

1. Dodržení standardního postupu nasazení prostředí
2. Zadání všech požadovaných parametrů dle specifikace /tzv. požadavky uživatele/
3. Soulad konfigurace HW a SW s definovanými standardy pro provoz UNIX/LINUX systémů
4. Soulad konfigurace HW a SW se všemi souvisejícími standardy /bezpečnost atd./

Podmínky obsažené v tomto dokumentu platí jak pro nově instalované, tak i všechny stávající servery a jejich periferie předávané do OCIS Infrastruktura – UNIX/LINUX systémy.

1.2 Rozsah působnosti

Postup je závazný pro zaměstnance organizačních jednotek OCIS a NAKIT, kteří požadují zajištění instalace, provozu a správy HW a OS, OCIS Infrastruktura – UNIX/LINUX.

1.3 Definice a termíny

Termín	Popis
Administrátor OS	Zodpovídá za instalaci, konfiguraci a správu HW a OS. Administrátor OS je vždy specialista OCIS/NAKIT.
Uživatel / žadatel	Uživatelem je myšlen buď přímý uživatel definovaný na úrovni OS, administrátor aplikace, administrátor databáze, přeneseně i projekt a externí dodavatel aplikace.
Aplikace	Aplikací se rozumí databáze, databázová úloha, aplikační server, web server, skript, spustitelný program, démon, job, procedura nebo jakákoliv jiná úloha, která potřebuje pro svůj běh prostředí s OS typu UNIX/LINUX.
OCIS Infrastruktura	Zahrnuje veškeré technické vybavení používané k provozu výpočetních systémů. Kromě fyzických zařízení – Hardware /serverů, síťových prvků a disk polí, .../ zahrnuje také podpůrná zařízení potřebná pro jejich provoz /systémy pro nepřetržité napájení, chlazení, dohled, .../

Termín	Popis
Hardware (HW)	Fyzická zařízení / komponenty výpočetní techniky, zahrnuje např. servery a jejich periferie (mechaniky CD/DVD, pásky, ...), disková pole, SAN a síťové přepínače atd.
Software (SW)	Souhrnný pojem pro veškeré programové vybavení instalované a používané na příslušném Hardware. Obsahuje: operační systémy, řešení pro správu (zálohování, dohled, vysoká dostupnost), databáze, aplikační programy, konfigurace, ...
Prostředí	Souhrnný pojem používaný pro skupinu Hardware a Software, které společně slouží provozu aplikací pro určený účel.
Neprodukční prostředí TEST	**TEST** (Testing) slouží primárně pro vývoj nových verzí aplikačního software. Pokud to konkrétní konfigurace dovoluje, je možné tato prostředí sdílet pro vývoj různých aplikací.
Produkční prostředí	**PROD** slouží pouze pro provoz aplikací, které mají produkční status, a proto zde nesmí běžet aplikace s jinou klasifikací /tzn. **Produkční prostředí nelze sdílet pro vývoj a testování aplikací! /. PROD** musí obsahovat pouze systémový SW a produkční aplikace pro daný IS. Přístupová práva do **PROD** musí být omezena na nezbytně nutná pro běh a kontrolu produkčních aplikací, pro instalaci patchů, a instalaci nových nebo opravených aplikačních programů. Veškeré změny produkčního **PROD** musí být před jejich provedením nejprve otestovány v **TEST**.
Cluster	V kontextu výpočetní techniky znamená skupinu spojených vzájemně spolupracujících serverů /tzv. cluster node/, jejichž účelem je společné poskytování výpočetních prostředků příslušné aplikaci. Jednotlivé node clusteru společně sdílejí některé prostředky /sítě, storage apod./. Minimální počet node v clusteru jsou 2.
Package	Označuje aplikační balíček a reprezentuje všechny zdroje, které aplikace potřebuje ke svému běhu v clusteru /konfiguraci IP sítě, diskové prostory /filesystémy apod./

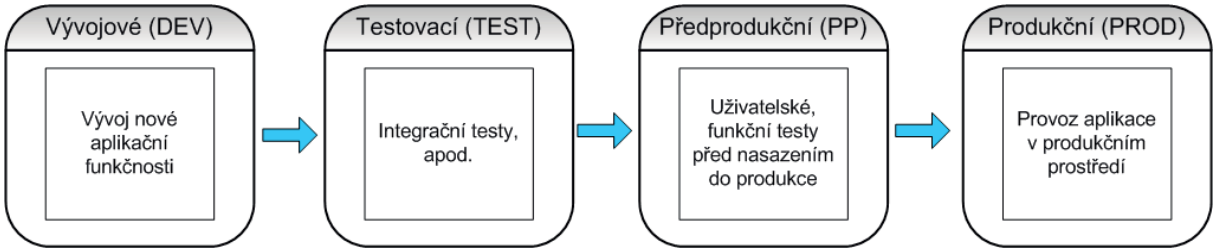
Tabulka 1 Definice a termíny

2 Implementace prostředí

Tato kapitola obsahuje základní informace o implementaci nového prostředí ve správě OCIS Infrastruktury – Oracle Solaris Sparc/Linux CentOS a Debian systémů do provozu.

2.1 Typy provozovaných prostředí

Následující obrázek dokumentuje zjednodušeně role jednotlivých prostředí ve vztahu k vývojovému cyklu aplikací.



Obrázek 1 Role provozovaných prostředí

2.2 Postup nasazení

Postup implementace nového prostředí má několik návazných fází, které stručně dokumentuje následující tabulka.

Č.	Popis fáze	Role uživatele / žadatele	Role OCIS	Výstup žadatel	Výstup OCIS
1	Definice a verifikace parametrů nového prostředí	Definice požadavků, zajištění podkladů, zadání RFC	Verifikace, případná korekce požadavků;	formulář požadavků	
2 a	Výběr HW/SW	-	Výběr z dostupných HW/SW prostředků vs. nový nákupů; FS	identifikace finančního krytí	Studie proveditelnosti
2 b	Pořízení nového HW	Zajištění krytí zdrojů: finanční + lidské	nákup a dodávka HW/SW, příprava pro instalaci HW	-	SLA s dodavatelem HW/SW
3	Instalace HW a OS	-	Instalace a konfigurace HW/OS	-	zařazení do knihy serverů
4	Pilotní provoz (stabilizace HW)	-	Kontrola funkce HW a OS	-	-
5	Impl. zálohování a dohledu OS	-	Implementace	-	zařazení do katalogu
6	Instalace aplikace	Instalace a konfig. aplikace, testy...	Podpora	N/A	-
7	Impl. zálohování a dohledu Aplikace	Zajištění podkladů	Implementace	formulář požadavků	zařazení do katalogu
8	Akceptace a převzetí provozu	zajištění podkladů	převzetí		akceptační protokol

Tabulka 2 Postup nasazení nového prostředí

2.2.1 Definice a verifikace parametrů nového prostředí

Prvním krokem k implementaci nového prostředí pro aplikaci je zadání uživatelských požadavků. Požadavky uživatele na různé parametry prostředí verifikuje a později realizuje administrátor OS. Všechny požadavky /parametry/ musí být v souladu se standardy.

Parametry, které nejsou uživatelem specifikovány, může Administrátor doplnit dle vlastních zkušeností s přihlédnutím k charakteru aplikace a jejím požadavkům, nebo může vyžadovat jejich doplnění uživatelem. V případě nedostatečných a nejasných podkladů má OCIS Infrastruktura právo zastavit implementaci do doby doplnění podkladů.

Výčet uživatelských požadavků /parametrů/ pro nastavení prostředí je obsažen v kapitole ***Chyba! Nenalezen zdroj odkazů. Chyba! Nenalezen zdroj odkazů.***.

Současně se zadáním parametrů nového prostředí uživatel zadává i požadavek na změnu /RFC/, který slouží pro krytí zdrojů /finančních a lidských/ příslušného požadavku. Obdržení RFC je pro OCIS Infrastrukturu podnětem pro zahájení procesu Studie proveditelnosti. Již v této fázi musí uživatel uvést požadovanou úroveň provozního zajištění /SLA/, od které se odvíjí celá řada parametrů prostředí.

2.2.2 Výběr a pořízení HW/SW

Po definici a schválení uživatelských požadavků na nové prostředí následuje v rámci Studie proveditelnosti výběr HW/SW, který má obvykle dvě možnosti:

1. Využití stávajícího HW v majetku MV ČR, který vyhovuje parametrům prostředí. V tomto musí být schváleno vlastníkem hw a udělena výjimka.
2. Výběr a nákup nového HW s odpovídajícími parametry

HW konfiguraci serveru musí žadatel navrhnout v písemné formě /viz předchozí krok/ a případně konzultovat se specialisty OCIS Infrastruktura. Schválenou konfiguraci potvrdí specialisté OCIS Infrastruktury také v písemné formě. Tyto kroky jsou součástí procesu Studie proveditelnosti.

OCIS Infrastruktura může s ohledem na různé požadavky provozu /dostupnost, redundanci, výkonnost/ navrhnout úpravy všech parametrů HW/SW serverů a storage:

- Platformu a model serveru, verzi a typ OS
- Model a počet CPU, velikost a typ paměti RAM
- Typ a konfiguraci I/O adapterů (síťové, diskové, jiné);
- Platformu, model a konfiguraci diskových prostorů
- Doplnění komponent (Rack Mount Kitu, redundantní zdroje a větráky, disky, ...)
- Navrhnout konsolidaci a virtualizaci celého prostředí, nebo jeho vybraných částí

Po výběru a schválení vhodného HW/SW následuje případný nákup HW a všech souvisejících systémových SW na náklady žadatele. Tento krok se řídí vnitřním předpisem MV ČR a NAKIT.

2.2.3 Instalace HW a OS

Instalaci HW do datového centra a instalaci jeho OS provádí specialisté OCIS/NAKIT. To zahrnuje následující kroky:

- Rozhodnutí o umístění HW do příslušného datového centra
- Zajištění požadavku na realizaci napájení a datových rozvodů v datovém centru
- Instalace HW do kabinetu 19“ v datovém centru
- Instalace a konfigurace OS dle standardů a požadavků uživatele + konfigurace HA
- Instalace SW podle požadavků uživatele
- Zajištění žádosti o záznam v DNS
- Konfigurace bezpečnostních nastavení /hardening OS/
- Základní infrastrukturní testy /vysoká dostupnost – boot, disky, sítě, cluster, .../

Tyto práce často obnáší objednání technologií či služeb dalších oddělení a tří i více zúčastněných stran a realizace instalace tak může trvat přibližně několik týdnů (4-15). Uživatel má po zadání požadavku nárok na vypracování hrubého časového odhadu instalace. Na vypracování časového odhadu má OCIS infrastruktura **3 pracovní dny** od doručení žádosti o jeho vypracování.

2.2.4 Pilotní provoz (stabilizace HW)

Před uvedením do produkčního provozu musí server běžet po dobu minimálně 14 dnů bez delšího přerušení (1 den). Během této doby lze provádět přípravu na spuštění prostředí do provozu (instalace a konfigurace aplikace, konfigurace svazků/disků, nastavení + ladění parametrů aplikace/OS atd.). Účelem pilotního provozu je prověření funkčnosti a stability všech HW i SW komponent serveru a operačního systému. V případě výskytu závad v pilotním provozu, je uvedení serveru do produkčního provozu posunuto o dobu nezbytně nutnou na vyřešení závad (oprava HW, řešení problému se SW apod.).

2.2.5 Implementace zálohování, monitoringu a dohledu OS

Implementace zálohování, monitoringu a dohledu OS je standardní krok, který provádějí administrátoři pro všechna prostředí bez výjimky. Řídí se dle související dokumentace.

2.2.6 Instalace aplikace (lze během pilotního provozu)

V této fázi je připravené prostředí (konfigurovaný HW a OS dle specifikace) dáno k dispozici uživateli, který instaluje, konfiguruje a testuje prostřednictvím příslušných uživatelských účtů aplikační software. V případě požadavku na součinnost administrátora musí uživatel předem o tuto předem požádat zodpovědného vedoucího.

2.2.7 Implementace zálohování, monitoringu a dohledu Aplikace

Zálohování, monitoring a dohled aplikací je samostatná úloha, která vyžaduje explicitní specifikaci uživatele.

V případě požadavku na provoz aplikace musí uživatel specifikovat následující parametry:

- Dokumentace správy logů aplikace (rotace, mazání, ...)
- Podklady pro zálohování aplikace
- Podklady pro nastavení sledování
- Stanovení administrátora aplikace (administrátor nesmí být z oddělení IT Infrastruktura)

2.2.8 Akceptace prostředí a převzetí serveru do provozu

K převzetí serveru do provozu dojde pouze tehdy, pokud je server po HW stránce řádně připraven s kompletně nainstalovaným OS, který je v souladu se standardy uvedenými v tomto dokumentu. Předpokladem pro produkční provoz aplikace je také zálohování serveru a dohled, viz související dokumenty.

Při přebírání serveru musí být podepsaná SLA (supportní smlouva) s dodavatelem HW & OS, nebo musí být uvedena neexistence podpory od dodavatele. SLA infrastruktury se odvíjí mimo jiné i od úrovně podpory dodavatele. SLA s dodavatelem HW & OS uzavírá oddělení obvykle při pořízení nového HW/OS.

Provoz akceptovaného prostředí se řídí provozní dokumentací.

3 Parametry provozovaných prostředí

Kapitola definuje kompletní seznam základních parametrů prostředí, které musí uživatel definovat pro každý systém předávaný do provozu /tzv. požadavky uživatele/.

Kapitola dále obsahuje podrobný výčet všech standardizovaných nastavení systémů s OS typu UNIX/LINUX, se kterými musí být v souladu všechna prostředí předávaná do provozu. Používání standardních nastavení není samoúčelné, vychází z provozních zvyklostí, best practices, doporučení dodavatelů a zejména podporuje klíčové požadavky OCIS provozu /efektivnost, bezpečnost, spolehlivost/.

3.1 Požadované parametry prostředí /uživatelské požadavky/

Prvním krokem k implementaci nového prostředí pro aplikaci je zadání požadavků /parametrů/ pro nastavení prostředí.

- Základní parametry prostředí jsou vstupem pro Studii proveditelnosti
- Detailní parametry OS jsou vstupem pro instalaci OS
- Zálohování aplikace /lze doplnit až při instalaci/
- Dohled aplikace /lze doplnit až při instalaci/
- Monitoring aplikace /lze doplnit až při instalaci/
- Diagram – Základní diagram/schéma aplikace

Zadavatel je povinen zadat parametry.

3.2 *Standardy pro Oracle Solaris Sparc/Linux CentOS a Debian*

Všechny parametry prostředí a v nich provozovaných aplikací musí odpovídat schváleným standardům a pravidlům, které shrnuje tato kapitola.

3.2.1 Podporované operační systémy

Podrobný výčet všech aktuálně podporovaných operačních systémů:

- UNIX Oracle Solaris 10 a vyšší
- LINUX RHEL a vyšší
- LINUX Centos 7.5
- LINUX Debian 10 a vyšší

3.2.2 Podporované virtualizační systémy Oracle Solaris Sparc/Linux CentOS a Debian

- LDOM na platformě Oracle Solaris Sparc
- RHEV/Ovirt
- VMware

3.2.3 Konfigurace HW

Vedle požadavků na typ platformy a OS musí HW integrovaný do infrastruktury splňovat následující požadavky.

- Konzole: karta/konzole pro vzdálenou správu serveru (iLO, RIB, ...)
- Mechaniky: DVD-ROM
- Rack-mount: všechny servery se umísťují do kabinetů 19“
- Rozšiřitelnost: server musí být rozšiřitelný (procesory, RAM, disky, IO karty).
- Napájení a chlazení: redundantní napájení (N+1)

3.2.4 Konfigurace OS

- Instalace OS probíhá podle instalačního postupu ze standardní instalační image
- Instalaci a konfiguraci OS provádí vždy administrátor/specialista OCIS/NAKIT
- Administrátor instaluje vždy aktuální a dostatečně stabilní verzi daného typu OS a jeho všech součástí a patch/repozitory. O výjimku (starší/novější verzi OS/patch/repozitory) musí uživatel explicitně požádat s doložením důvodů (např. certifikovaná platforma aplikace)

3.2.5 Minimální konfigurace pro virtuál Oracle Solaris Sparc

- minimálně 1x dualcore
- 16 GB RAM per CPU core
- 2x virtuál disk (minimálně 128 GB)

3.2.6 Minimální konfigurace pro virtuál Linux CentOS a Debian

- minimálně 2 vCPU
- 4 GB RAM per CPU core
- 1x virtuál disk (minimálně 64 GB)

3.2.7 Dohled, monitoring a zálohování

- Na všech systémech ve správě specialistů OCIS/NAKIT jsou před zařazením do provozu konfigurovány úlohy pro pravidelnou zálohu, monitoring a dohled OS. Preferovaný systém pro monitoring je Zabbix, který funguje jako distribuovaný monitoring pomocí Zabbix proxy, dedikovaných pro jednotlivá DC. Preferovaný systém pro dohled je Graylog, který funguje jako distribuovaný dohled pomocí Gralog proxy, dedikovaných pro jednotlivá DC.

3.2.8 Konfigurace sítě

- Nastavení vysoké dostupnosti sítě (bonding ...) je mandatorní pro všechna prostředí typu Mission Critical a Critical; Doporučeno je i pro ostatní.
- Hostname serveru vychází z následující jmenné konvence **SSAATYP00** kde:
  SS … typ a OS serveru
  AA … zkratka aplikačního celku
  TYP … zkratka využití serveru

  Povolené parametry jsou uvedené v následující tabulce.

Významová tabulka jednolivých částí jménné konvence *)							
	Typ zařízení/rozhranní		Aplikační celek		Bližší určení		Číslo instance bližšího určení
LS	Linux server	SD	Service desk	DBS	DB server	01 až 99	Vypňuje se vždy - určuje pořadové číslo např. diskového pole č. 4, DB server č. 5, rack server č. 8 apod.
SS	Solaris Server	EK	EKIS	APL	Apl. Server		
NS	Windows server (NT server)	EG	DCeGOV	WEB	Web server		
LV	Linux virtual	CM	CMS2	DNS	DNS server		
SV	Solaris virtual	IS	ISOSS	NTP	NTP server		
NV	Windows virtual (NT virtual)	IN	INFRA	LDP	LDAP server		
		FR	FRS	DHC	DHCP server		
				BCK	Backup server		
				ARC	Archivační server		
				LTO	LTO server		
				NFS	NFS server		
				RHV	Server s virtualizací RHEV		
				LDM	Server s virtualizací Ldom		
				HYV	Server s virtualizací Hyper-V		
				ZBX	Server dohled Zabbix		
				SCM	Server dohled SCOM		
				FTP	FTP server		

Tabulka 3 Významová tabulka jmenné konvence

Příklady tvorby jmen serverů:

Příkladová tabulka jmenné konvence					
				Výsledné Jméno	Poznámka
LV	EK	NTP	01	LVEKNTP01	Virt. server , OS Linux, DB server pro EKIS
LV	EK	DBS	02	LVEKDBS02	Virt. server , OS Linux, NTP služba pro EKIS
LS	SD	BCK	02	LSSDBCK02	Fyzický server , OS Linux, backup pro ServiceDesk
NS	SD	HYV	03	NSSDHYV03	Fyzický server , OS Windows, Hyper-V server pro ServiceDesk
SV	SD	APL	04	SVSDAPL04	Virt. server , OS Solaris, APL server pro ServiceDesk
LV	SD	DNS	01	LVSDDNS01	Virt. server , OS Linux, DNS pro ServiceDesk
LV	IN	ZBX	01	LVINZBX01	Virt. server , OS Linux, infrastrukturní dohledový server Zabbix

Tabulka 4 Hostname pro UNIX/Linux servery

- Hostname serveru se nastavuje jako A-záznam k primárnímu síťovému interface
- Všechny servery se při instalaci zařadí do příslušné domény DNS
- Ve všech konfiguracích aplikací obsahujících adresy v IP síti je bezpodmínečně nutné uvádět pouze FQDN. **Je zakázáno používání krátkých jmen a zejména IP adres!**
- Do DNS se zavádějí také všechny adresy package v Clusteru. Uživatel musí používat příslušné FQDN pro přístup k aplikaci běžící v clusteru po IP síti.

3.2.9 Konfigurace storage

- Interní disky v serveru včetně systémových musí být mít ochranu RAID (možno realizovat prostředí HW řadiče nebo prostř. SW)
- Všechny externí diskové prostory musí mít ochranu. Typ ochrany (RAID) a rozdělení diskových prostorů určuje výhradně administrátor.
- Podporovány jsou pouze diskové systémy umožňující bezodstávkový provoz 24x7 (redundantní napájení, upgrade HW / FW / utilit za běhu)
- Dokupované diskové prostory musí být v násobcích nejmenší možné kapacity disků a násobky musí odpovídat nárokům na optimální rozložení a požadovaný výkon
- Zapojení serverů do SAN musí být redundantní /2 a více nezávislých FC kanálů/
- Aplikace nesmí být instalovány na systémových partitions (svazcích). Výjimku tvoří ryze systémové programy, např. DNS, SSHD, ...
- Konfiguraci souborového systému (striping, direct I/O, mount point apod.) navrhuje a realizuje administrátor na základě podkladů od uživatele a dle best practices

3.2.10 Parametry dostupnosti prostředí

Způsob zajištění vysoké dostupnosti /HA/ a obnovy po havárii většího rozsahu /Disaster Recovery/ vychází z požadavků na dostupnost daného prostředí, která se určuje dle kategorie aplikace (stupnice kritičnosti).

Kategorie kritičnosti je stanovena vnitřním předpisem**Chyba! Nenalezen zdroj odkazů.** a má 2 stupně:

- KIS Kritický informační systém
- VIS Významný informační systém

Související pravidla dostupnosti prostředí:

- Konfigurace HW a SW a způsob zajištění HA/DR každého prostředí musí odpovídat požadavkům na jeho dostupnost (kritičnost)
- Dostupnost celého prostředí přímo závisí na jeho nejslabším článku, čemuž musí odpovídat požadovaná SLA. Výjimku tvoří clusterová řešení, která eliminují některé SPOF.
- Úrovni kritičnosti prostředí musí odpovídat i servisní zajištění provozovaného HW a SW (SLA s dodavatelem). Úroveň HW podpory externího dodavatele musí být v harmonii v požadovanou SLA
- Clusterová řešení musí odpovídat seznamu podporovaných platforem

3.2.11 Bezpečnost prostředí a přístupová práva

- Administrátorem HW a OS a vlastníkem superuživatele /root/ musí být pracovník/specialista OCIS/NAKIT. Účet root je zakázáno sdílet s uživatelem.
- Pokud aplikace při instalaci vyžaduje administrátorská práva (root, sys …), musí instalaci provádět administrátor OS ve spolupráci s uživatelem
- Aplikace nesmí běžet pod superuživatelem (root). Výjimku tvoří systémové programy, např. DNS, SSHD, ... Pro možnost využívání vyšších práv v OS UNIX se používá software SUDO, který umožňuje řízené a omezené propůjčení identifikace

- Pro správu jsou povolené pouze zabezpečené komunikační protokoly (SSH, SFTP, HTTPS apod.). **Používání následujících služeb není dovoleno z důvodu bezpečnosti a spolehlivosti: NFS, SAMBA, sendmail, telnet, remsh utility, rpcbind, FTP**
- Každý OS musí být převeden do chráněného režimu (trusted a audit mode, shadow files, ...)
- Patch/repozitory a bezpečnostní záplaty jsou aplikovány na všech OS po jejich uvedení do provozu v pravidelných intervalech na základě aktuálně používané patchové analýzy
- Vytváření uživatelských identifikací do operačního systému zajišťuje administrátor OS po zadání požadavku uživatele.
- Přímý přístup do OS neboli uživatel s přístupem k zadávání příkazů OS, se přiděluje pouze v odůvodněných případech
- HW se umísťuje do datových center (zabezpečených objektů a místností) ve správě OCIS Infrastruktury za asistence administrátora HW/OS.
- Fyzický přístup k serveru má pouze administrátor HW/OS
- Do datových center nelze umísťovat zařízení, která nejsou ve správě specialistů OCIS/NAKIT Infrastruktury.
- Povolení vzdáleného přístupu (VPN mezi serverem a sítí externího dodavatele) musí být schváleno odborem kybernetické bezpečnosti.

3.2.12 Řešení výjimek, odchylek od standardního nastavení

- Veškeré požadavky na výjimky ze standardů OCIS infrastruktury je nutné předem řešit s vedoucím, případně managerem infrastruktury
- Veškeré požadavky na výjimky z bezpečnostních nastavení OS je nutné předem řešit s odborem kybernetické bezpečnosti
- O instalaci a konfiguraci systémového SW rozhoduje vždy administrátor, který je odpovědný za správnou funkci OS a jeho provoz. V případě kolizních požadavků, pro které nelze splnit uvedenou podmínku, může tyto zamítnout (například odinstalování důležitých systémových patchí, nestandardní rekompilace části OS atd.).

3.3 *Odpovědnosti a pravomoci*

Implementace každého nového prostředí do OCIS Infrastruktury musí probíhat dle postupů stanovených tímto dokumentem.

3.3.1 OCIS Infrastruktura

Instalace, správa a provoz HW s OS Oracle Solaris Sparc/Linux CentOS a Debian jsou v odpovědnosti OCIS Infrastruktura. Osoba odpovědná za chod a dodržování schválených standardů OCIS Infrastruktury je vedoucí oddělení.

Veškeré změny a návrhy nových standardů OCIS Infrastruktury podléhají schválení útvarem Hlavního architekta eGovernmentu odbor Hlavního architekta Ministerstva vnitra a oddělením Provozu OCIS Infrastruktury. Bez schválení vedoucího OCIS Infrastruktury, útvaru Hlavního architekta eGovernmentu a Managera OCIS Infrastruktury nelze implementovat nové řešení OCIS infrastruktury, které odporuje standardům!

3.3.2 Administrátor

Administrátor OS má úplný přístup ke všem parametrům a zdrojům operačního systému. Zodpovídá za nastavení HW a OS, jeho běh, zálohu a obnovu OS, monitorování HW zdrojů, zatížení OS i chybových stavů. V případě výskytu chybových stavů HW nebo OS odpovídá za jejich řešení a případnou eskalaci na dodavatele.

Administrátor má právo všechny požadavky na konfiguraci infrastruktury odporující standardům odmítnout nebo upravit. Každá případná změna je oznámena uživateli, včetně jejího stručného zdůvodnění.

Administrátor má právo s ohledem na parametry dostupnosti, spolehlivosti, výkonnosti a správy prostředí průběžně provádět změny různých systémových nastavení HW/OS:

- Konsolidace/virtualizace/upgrade HW a OS, úprava parametrů OS (např. jádra); instalace patch, změna konfigurace podpůrných SW; úprava konfigurace storage apod.
- Pravidelné Testování funkčnosti HA clusterů a DR řešení /v součinnosti s uživatelem/

Administrátor má povinnost informovat uživatele o změnách v prováděných v konfiguraci HW a OS, které mohou mít vliv na provoz prostředí a aplikací.

3.3.3 Uživatel /žadatel/

Uživatel je povinen dodržovat všechny standardy OCIS Infrastruktury stanovené tímto dokumentem spolu se souvisejícími standardy /bezpečnost apod./.

Uživatel (žadatel o přípravu prostředí) je povinen zadávat požadavky na nastavení prostředí a procedur spojených s během aplikace ve formě stanovené tímto dokumentem. Dojde-li ke změně požadavků a nastavení prostředí na straně uživatele, musí uživatel neprodleně změnu s administrátory konzultovat.

S ohledem na kapacitní plánování musí uživatel vždy konzultovat s administrátorem všechny plánované změny, které mohou mít dopad na výkon/zatížení systému, databáze, disků /např. plánovaný nárůst zákazníků, významná změna funkčnosti aplikace apod./.

4 Přílohy

4.1 Zkratky

Zkratka	Popis
CMDB	Change Management Database
DMZ	Demilitarized Zone
DNS	Domain Name System
DR	Disaster Recovery
FC	Fibre Channel
FQDN	Fully Qualified Domain Name /např. hostname.nakit.cz /
HA	High Availability
IP	Internet Protocol
RAID	Redundant Array of Independent Disks
RFC	Request For Change
SAN	Storage Area Network
SLA	Service Level Agreement
SPOF	Single Point Of Failure
SUDO	SuperUser „DO“ – program pro elevaci práv v OS

5 Přílohy

5.1 Zkratky

Zkratka	Popis
CMDB	Change Management Database
DMZ	Demilitarized Zone
DNS	Domain Name System
DR	Disaster Recovery
FC	Fibre Channel
FQDN	Fully Qualified Domain Name /např. hostname.nakit.cz /
HA	High Availability
IP	Internet Protocol
RAID	Redundant Array of Independent Disks
RFC	Request For Change
SAN	Storage Area Network
SLA	Service Level Agreement
SPOF	Single Point Of Failure
SUDO	SuperUser „DO“ – program pro elevaci práv v OS

Tabulka 5 Zkratky