О результатах моделирования управления доступом в СУБД PostgreSQL в рамках МРОСЛ ДП-модели

чл.-корр. АК России,
д.т.н., профессор Девянин П.Н.

НАУЧНО-ПРОИЗВОДСТВЕННОЕ ОБЪЕДИНЕНИЕ
РУСБИТЕХ

Требования по безопасности информации, устанавливающие 6 уровней доверия (приказ ФСТЭК России от 30.07.2018 № 131)

5 уровень доверия (объекты КИИ 2 категории, ГИС 2 класса защищенности)

- идентификация и анализ скрытых каналов по памяти.

4 уровень доверия (объекты КИИ 1 категории, ГИС 1 класса защищенности)

- модель безопасности, включая реализуемые политики управления доступом и фильтрации информационных потоков.

3 уровень доверия (ИС, в которых обрабатывается информация, содержащая секретные сведения)

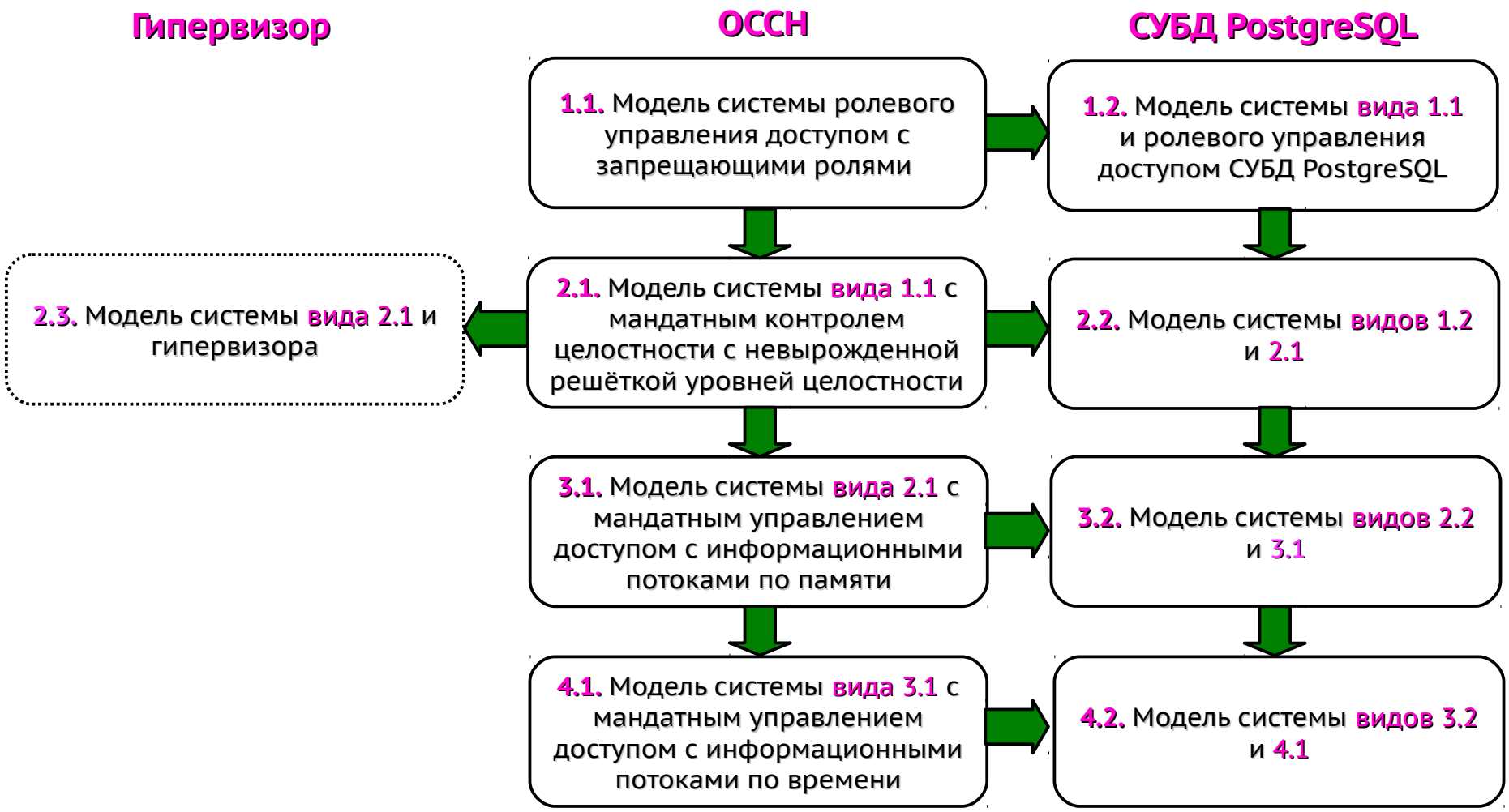
- верификация модели безопасности с использованием инструментальных средств;
- идентификация и анализ скрытых каналов по времени.

1 уровень доверия (ИС, в которых обрабатывается информация, содержащая сведения особой важности)

- идентификация и анализ скрытых статистических каналов.

ГОСТ Р «Защита информации. Формальное моделирование политики безопасности. Часть 1. Формальная модель управления доступом» (проект);
ГОСТ Р «Защита информации. Формальное моделирование политики безопасности. Часть 2. Верификация формальной модели управления доступом» (проект)

Актуальное иерархическое представление МРОСЛ ДП-модели. Перспективы развития



Учет специфики СУБД *PostgreSQL*

Роли СУБД

$DB\_ADMIN\_PRIVILEGES = \{SUPERUSER, CREATEROLE, CREATEDB, LOGIN, REPLICATION, INHERIT\}$ – множество административных привилегий СУБД;

$DB\_AR \subset DB\_R$ – множество административных ролей СУБД;

$DB\_SR \subset DB\_R$ – множество специальных ролей СУБД ($postgres\_admin\_role \in DB\_SR \cap DB\_AR$ – специальная административная роль для учёта привилегии *SUPERUSER*);

$DB\_COMMON \subset DB\_R$ – множество общих ролей СУБД (для учёта атрибута *PUBLIC*);

$DB\_AP: DB\_R \rightarrow 2^{DB\_ADMIN\_PRIVILEGES}$ – функция административных привилегий ролей СУБД;

$db\_login: S \rightarrow \{r \in DB\_R : LOGIN \in DB\_AP(r)\}$ – функция роли входа субъект-сессии в СУБД (для учета привилегии *LOGIN*);

$db\_inherit: DB\_R \rightarrow 2^{DB\_R}$ – функция наследования привилегий ролей к элементам СУБД (для учета привилегии *INHERIT*);

$db\_with\_admin\_option: DB\_R \rightarrow 2^{DB\_R}$ – функция управления подчинённостью ролей в иерархии (для учёта атрибута *WITH ADMIN OPTION*);

$DB\_APA: AR \cup DB\_AR \rightarrow 2^{DB\_R \times R_r}$ – функция административных прав доступа административных ролей ОССН и СУБД к ролям СУБД.

Учет специфики СУБД *PostgreSQL*

Элементы СУБД

$DB\_E = DB\_O \cup DB\_C$ – элементы СУБД, не являющихся ролями, где DB_O – элементы-объекты СУБД (расширения, сопоставления, домены, конфигурации, словари, парсеры, шаблоны, функции, последовательности, строки, ограничения, индексы, правила, триггеры, триггерные функции, репликации), DB_C – элементы-контейнеры СУБД (кластеры, базы данных, схемы, каталоги, таблицы, столбцы, представления, табличные пространства) и по определению $DB\_O \cap DB\_C = \varnothing$, $DB\_O \cap C = \varnothing$, $DB\_C \cap O = \varnothing$;

$DB\_PRIVILEGES = \{SELECT, INSERT, UPDATE, DELETE, TRUNCATE, REFERENCES, TRIGGER, USAGE, CREATE, CONNECT, TEMPORARY, TEMP, EXECUTE, OWN\}$ – виды привилегий СУБД;

$E \cap DB\_E$ – сущности СУБД (кластеры, базы данных, каталоги, расширения, схемы, репликации, табличные пространства). При этом по определению существуют единственная сущность-контейнер СУБД $DB\_CLUSTER \in C \cap DB\_C$, являющаяся кластером СУБД, и множество сущностей-контейнеров $DB\_SCHEMES \subset C \cap DB\_C$, являющихся схемами (каталогами) СУБД;

$DB\_P \subseteq DB\_E \times DB\_PRIVILEGES$ – множество привилегий к элементам СУБД;

$db\_privileges: DB\_R \rightarrow 2^{DB\_P \times DB\_R}$ – функция привилегий к элементам СУБД ролей СУБД (для учёта права *WITH GRANT OPTION*);

$DB\_PA: DB\_R \rightarrow 2^{(E \cap DB\_E) \times R_r}$ – функция эффективных прав доступа ролей СУБД.

Уровень 2.2. Пример де-юре правила преобразования состояний

		access_read(x, x', y, α_p)	
1.1	x, y	$x \in S$, если $y \in E \cup R \cup NR \cup AR$, то существует $r \in R \cup AR$: $(x, r, read_a) \in AA$, [если $y \in E$, то $(y, read_r) \in PA(r)$ и существует контейнер $c \in C$ такой, что $execute\_container(x, c, y) = true$, и не существует запрещающей роли $nr \in NR$ такой, что $(x, nr, read_a) \in AA$ и $(y, read_r) \in PA(nr)$], [если $y \in R \cup NR \cup AR$, то $(y, read_r) \in APA(r)$], [если $y \in R \cup AR$, то для всех $nr \in constraint_{NR}(y)$ верно $(x, nr, read_a) \in AA$]	если $y \in E$, то $A' = A \cup \{(x, y, read_a)\}$, если $y \in R \cup NR \cup AR$, то $AA' = AA \cup \{(x, y, read_a)\}$
1.2	x'	$x' \in S$, [если $y \in R \cup NR \cup AR$, то $i_r(y) \le i_s(x)$, для $e \in]y[$ либо $(x, e, read_a) \in A$, либо $(x, e, write_a) \in A$], [если $y \in R \cup NR \cup AR$ и $i_r(y) > i\_low$, то $(x', i\_entity, write_a) \in A$]	–
2.1	α_p	[если $y \in E \setminus DB\_E$, то $\alpha_p = \varnothing$], [если $y \in DB\_E$, то или $(x, postgres\_admin\_role, read_a) \in AA$, или существует $r \in DB\_R$: $(x, r, read_a) \in AA$, $\alpha_p \in DB\_PRIVILEGES$, $read_r \in db\_rights(\alpha_p)$, $(db\_entity(y), read_r) \in DB\_PA(r)$, и существует контейнер $c \in C \cup DB\_C$ такой, что $execute\_container(x, c, y) = true$], [если $y \in DB\_R$, то $y \ne public\_role$, $\alpha_p = \varnothing$ и или [$db\_login(x) = \varnothing$, существует $r \in AR$: $(x, r, read_a) \in AA$, $(y, read_r) \in DB\_APA(r)$], или [$(x, postgres\_admin\_role, read_a) \in AA$ или $y \le db\_login(x)$]]	если $y \in DB\_E$, то $A' = A \cup \{(x, db\_entity(y), read_a)\}$, если $y \in DB\_R$, то если $db\_login(x) = \varnothing$, то [$db\_login'(x) = y$, $AA' = AA \cup \{(x, y, read_a), (x, public\_role, read_a)\}$], иначе [$AA' = (AA \cup \{(x, y, read_a)\}) \setminus \{(x, y', read_a) \in AA : y' \in DB\_R \setminus \{public\_role, y\}\}$]
2.2	–	если $y \in DB\_R$, то [для $e \in]y[$ либо $(x, e, read_a) \in A$, либо $(x, e, write_a) \in A$], [$i_r(y) \le i_s(x)$, и если $y \ne public\_role$, то $i_r(y) = i_r(db\_login(x))$], [если $i_r(y) > i\_low$, то $(x', db\_i\_entity, write_a) \in A$]	–

Уровень 4.2. Пример де-факто правила преобразования состояний

	post(*x*, *y*, *z*)	
2.1	$x, z \in S, y \in (E \setminus E\_HOLE), x \neq z,$ $(x, y, write_m) \in F,$ $(y, read_a) \in de\_facto\_accesses(z)$	$F' = F \cup \{(x, z, write_m)$
3.1	–	–
4.1	$(x, y, \alpha_f) \in F$, где $\alpha_f \in \{write_m, write_t\}$, [если $y \in (E \setminus RW\_HOLE) \cup R \cup NR \cup AR$, то $(y, \beta_a) \in de\_facto\_accesses(z)$, где $\beta_a \in R_a$)], [если $y \in S$, то $y \in de\_facto\_own(z)$]	–
4.2	если $y \in DB\_R$, то $(y, \beta_a) \in de\_facto\_accesses(z)$, где $\beta_a \in R_a$	если $\alpha_f = write_m$, $\beta_a = read_a$ и $y \in E \setminus E\_HOLE$, то $F' = F \cup \{(x, z, write_m)\}$, иначе если $x, z \in N_S \cup NF_S$ и [если $y \in (E \cap DB\_E) \cup DB\_R$, то $x, z \notin DB\_LF_S$], то $F' = F \cup \{(x, z, write_t)\}$

Безопасность системы

Пусть G_0 – безопасное начальное состояние системы $\Sigma(G^*, OP, G_0)$, и существует траектория без кооперации доверенных и недоверенных субъект-сессий $G_0 \vdash_{op1} G_1 \vdash_{op2} \dots \vdash_{opN} G_N$, где $N \geq 1$.

2 уровень. Безопасность в смысле мандатного контроля целостности

Определение о.Ц.07. Будем говорить, что в состоянии G_N произошло нарушение безопасности системы в смысле мандатного контроля целостности, когда существуют недоверенная субъект-сессия $x \in N_{SN}$ и доверенная субъект-сессия $y \in de\_facto\_own_N(x) \cap L_{SN}$ такие, что не верно неравенство $i_s(y) \leq i_s(x)$, и это условие не выполняется в состояниях G_i траектории, где $0 \leq i < N$. Назовём систему $\Sigma(G^*, OP, G_0)$ безопасной в смысле мандатного контроля целостности, когда в ней невозможно соответствующее нарушение безопасности.

3 уровень. Безопасность в смысле Белла-ЛаПадулы

Определение о.КП.01. Будем говорить, что в состоянии G_N произошло нарушение безопасности системы в смысле Белла-ЛаПадулы, когда существует информационный поток по памяти $(x, y, write_m) \in F_N$ такой, что $x, y \in E_N$ и не верно неравенство $f_{eN}(x) \leq f_{eN}(y)$, и это условие не выполняется в состояниях G_i траектории, где $0 \leq i < N$. Назовём систему $\Sigma(G^*, OP, G_0)$ безопасной в смысле Белла-ЛаПадулы, когда в ней невозможно соответствующее нарушение безопасности.

4 уровень. Безопасность в смысле контроля информационных потоков по времени

Определение о.КВ.02. Будем говорить, что в состоянии G_N произошло нарушение безопасности системы в смысле контроля информационных потоков по времени, когда в нём существует информационный поток по времени $(x, y, write_t) \in F_N$ такой, что $x, y \in E_N$ и не верно неравенство $f_{eN}(x) \leq f_{eN}(y)$, и это условие не выполняется в состояниях G_i траектории, где $0 \leq i < N$. Назовём систему $\Sigma(G^*, OP, G_0)$ безопасной в смысле контроля информационных потоков по времени, когда в ней невозможно соответствующее нарушение безопасности.

Уровень 3.2. Безопасность в смысле Белла-ЛаПадулы

Теорема т.КП.01.БДКП. Пусть G_0 – безопасное начальное состояние системы $\Sigma(G^*, OP, G_0)$. Пусть на всех траекториях системы без кооперации доверенных или недоверенных субъект-сессий $G_0 \vdash_{op1} G_1 \vdash_{op2} \dots \vdash_{opN} G_N$, где $N \geq 0$, и в каждом состоянии G_N для каждой субъект-сессии $s \in S_N$ и сущности $e \in E_N$ выполняются следующие условия.

Условие Ц.1.КП.БДКП. (корректность уровней целостности и конфиденциальности сущностей, функционально ассоциированных с субъект-сессиями) Если $e \in [s]$, то выполняются условия $i_{sN}(s) \leq i_{eN}(e)$ и ($f_{sN}(s) = f_{eN}(e)$ или [если $e \in E \setminus DB\_E$, то $i_{eN}(e) = i\_high$, иначе $i_{eN}(e) = db\_i\_high$]).

Условие Ц.2.КП.БДЦ.БДКП. (корректность уровней целостности и конфиденциальности, а также прав доступа на чтение к сущностям, параметрически ассоциированным с субъект-сессиями) Если $e \in]s[$, то верно $f_{sN}(s) = f_{eN}(e)$, $i_{sN}(s) \leq i_{eN}(e)$ и [для каждой роли или административной роли и $r \in R_N \cup AR_N$ такой, что $(e, read_r) \in PA_N(r)$], [для каждой роли СУБД $r \in DB\_R_N$ такой, что $(e, read_r) \in DB\_PA_N(r)$], выполняется условие $i_{eN}(e) \leq i_{rN}(r)$.

Условие Ц.3.КП. (без дополнений).

Условие КП.4. (без дополнений).

Условие БДЦ.4. (без дополнений).

Тогда на этих траекториях система $\Sigma(G^*, OP, G_0)$ безопасна в смысле мандатного контроля целостности и Белла-ЛаПадулы.

Спасибо за внимание!