Пулеко І. В., к.т.н., доцент,
Державний університет «Житомирська політехніка»
Іщенко І. А., ст. викладач,
Свистунович І. В., викладач,
Житомирський військовий інститут ім. С. П. Корольова

ДЕТЕКТОР ВТОРГНЕНЬ ДЛЯ «РОЗУМНОГО» БУДИНКУ

Технологія «розумного» будинку вже давно перестала бути фантастикою. Сьогодні «розумний» будинок (розумний дім/smarthome, digitalhouse) − це система домашніх пристроїв, здатних виконувати дії і вирішувати певні повсякденні завдання без участі людини. Тут функціонально пов'язуються між собою усі електроприлади будівлі, якими можна керувати централізовано − з пульта-дисплею. Прилади, частіше всього, під'єднані до комп'ютерної мережі, що дозволяє керувати ними за допомогою комп'ютерної техніки та надає віддалений доступ до них через Інтернет. Завдяки інтеграції інформаційних технологій у домашні умови, усі системи та прилади узгоджують виконання функцій між собою, виконуючи задані програми відповідно до зовнішніх показників.

Попри прогрес у технологіях розумного будинку, проблеми кібербезпеки залишаються актуальними. Зловмисники можуть використовувати централізоване або віддалене керування для своїх цілей. Нове шкідливе програмне забезпечення, як, наприклад, програми-вимагачі, може шкодити користувачам, вимагаючи викуп за повернення контролю. Отримавши доступ до системи, зловмисники можуть керувати різними пристроями, включаючи опалення, кондиціонер, освітлення та систему безпеки.

Сьогодні безпека «розумного» будинку – це сектор інформаційних технологій, який фокусується на захисті кінцевих пристроїв, мереж та даних, а саме підключених пристроїв, які не є комп'ютерами, смартфонами або планшетами. По суті, безпека «розумного» будинку – це широкий термін, що охоплює стратегії, політики, процеси та технології безпеки, які використовуються для захисту своїх пристроїв та пов'язаних з ними даних, додатків та мережі від кіберзагроз, злому або іншої шкоди.

Перспективи розвитку таких загроз у майбутньому залежатимуть від дій правоохоронців та серйозності покарання зловмисників, кількості жертв, які готові платити кіберзлочинцям, та наявності потенційно цікавих для зловмисників цілей. Тому тема роботи, що направлена на виявлення порушень у мережі «розумного» будинку шляхом розробки детектора вторгнень є актуальною.

В аналізі даних є два напрямки, які займаються пошуком аномалій: детектування викидів (OutlierDetection) та «новизни» (NoveltyDetection). Як і викид "новий об'єкт" – це об'єкт, який відрізняється за своїми властивостями від об'єктів (навчальної) вибірки. Але на відміну від викиду, його в самій вибірці поки немає (він з'явиться через деякий час, і завдання якраз і полягає в тому, щоб виявити його з появою). Наприклад, якщо ми аналізуємо виміри температури у кімнаті та відкидаємо аномально великі чи маленькі, то ми боретеся з викидами. А якщо ми створюємо алгоритм, який для кожного нового виміру стану мережі оцінює, наскільки він схожий на минулі, і виділяє аномальні – ми шукаємо новизну (по суті створюємо детектор вторгнень).

Завдання детектування вторгнень немає єдиного формулювання і найчастіше інтерпретуються по-різному залежно від характеру даних і поставленої мети. Новизна, як правило, з'являється в результаті нової поведінки об'єкта. Скажімо, якщо наші об'єкти – опис роботи системи, то після проникнення в неї вірусу об'єкти стають «новизною». Тут важливо розуміти, що «новизна» називається новизною з тієї причини, що такі описи для нас абсолютно нові, а нові вони тому, що ми не можемо в навчальній вибірці мати інформацію про всілякі зараження вірусами або всілякі поломки. Формування такої навчальної вибірки трудомістке і часто не має сенсу. Проте можна набрати досить велику вибірку прикладів нормальної (штатної) роботи системи. Таким чином, детектор вторгнень буде являти собою програму навчену виявляти відхилення від нормальної роботи мережі «розумного» будинку.

Для реалізації програми можуть бути застосовані численні пакети виявлення викидів, що існують у різних мовах програмування. На думку авторів слід використовувати мову програмування Python та спеціалізовану бібліотеку PyOD.

PyOD – це масштабований набір інструментів Python для виявлення викидів у багатовимірних даних. Він надає доступ до більш як 20 алгоритмів виявлення викидів у рамках єдиного добре задокументованого API. Основними перевагами PyOD є наступні:

- Відкритий код із документацією та прикладами алгоритмів.
- Підтримує розширені моделі, включаючи нейронні мережі, глибоке навчання та ансамблі методів.
- Оптимізована продуктивність за допомогою розпаралелювання з використанням numba і joblib.
- Бібліотека сумісна як з Python 2, так і з 3.