
**KAUNO TECHNOLOGIJOS UNIVERSITETAS**

**ELEKTROS IR ELEKTRONIKOS FAKULTETAS**

**Valdemaras Tubutis**

# SUPERVIZORINIO VALDYMO IR DUOMENŲ SURINKIMO TAIKOMŲJŲ PROGRAMŲ APSAUGOS POSISTEMIŲ TYRIMAS

Baigiamasis magistro projektas

**Vadovas**
Doc. dr. Leonas Balaševičius

**KAUNAS, 2017**

**KAUNO TECHNOLOGIJOS UNIVERSITETAS**
**ELEKTROS IR ELEKTRONIKOS FAKULTETAS**
**AUTOMATIKOS KATEDRA**

# SUPERVIZORINIO VALDYMO IR DUOMENŲ SURINKIMO TAIKOMŲJŲ PROGRAMŲ APSAUGOS POSISTEMIŲ TYRIMAS

Baigiamasis magistro projektas
**Valdymo technologijos (kodas 621H66001)**

**Vadovas**
Doc. dr. Leonas Balaševičius

**Recenzentas**
Dr. Virginijus Baranauskas

**Projektą atliko**
Valdemaras Tubutis

**KAUNAS, 2017**

KAUNO TECHNOLOGIJOS UNIVERSITETAS

Elektros ir elektronikos

(Fakultetas)

Valdemaras Tubutis

(Studento vardas, pavardė)

**Valdymo technologijos (kodas 621H66001)**

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Supervizorinio valdymo ir duomenų surinkimo taikomųjų programų apsaugos posistemių tyrimas“

**AKADEMINIO SĄŽININGUMO DEKLARACIJA**

20 17 m. 06 01 d.

Kaunas

Patvirtinu, kad mano **Valdemaro Tubučio** baigiamasis projektas tema „Supervizorinio valdymo ir duomenų surinkimo taikomųjų programų apsaugos posistemių tyrimas“ yra parašytas visiškai savarankiškai, o visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

*(vardą ir pavardę įrašyti ranka)* *(parašas)*

Valdemaras Tubutis. Supervizorinio valdymo ir duomenų surinkimo taikomųjų programų apsaugos posistemių tyrimas. Valdymo sistemų *Magistro* baigiamasis projektas / vadovas doc. dr. Leonas Balaševičius; Kauno technologijos universitetas, Elektros ir elektronikos fakultetas, Automatikos katedra.

Mokslo kryptis ir sritis: Elektros ir elektronikos inžinerija, Technologiniai mokslai

Reikšminiai žodžiai: *SCADA, saugumas, vartotojai.*

Kaunas, 2017. 44 p.

## SANTRAUKA

Šiame darbe apžvelgiamos supervizorinio valdymo ir duomenų surinkimo taikomųjų programų apsaugos posistemių vartotojų autoretizavimo priemonės bei vartotojų funkcijų ribojimas. Nagrinėjami trys pagrindiniai posistemiai – WinCC, InTouch ir CitectSCADA. Taip pat apžvelgiamos galimos tinklo lygmens atakos prieš SCADA sistemas, ir bandoma pagrįsti, kodėl beveik visas dėmesys skiriamas tinklo saugumui. Atsižvelgiant į tai, kad pagrinde dėmesys skiriamas tinklo lygmeniui, nutarta ištirti vartotojų autoretizavimo saugumą. Saugumas išnagrinėtas tiek aplikacijos, tiek operacinės sistemos lygmenyje. Taip pat nustatyta, kokią įtaką turi projekto bei pačios programos vykdomųjų bylų trynimas. Visas tyrimas atliekamas turint tik operacinės sistemos administratoriaus teises.

Nagrinėjant saugumą aplikacijos lygmenyje, buvo bandoma nustatyti, ar turint sistemos administratoriaus teises galima pridėti naujus vartotojus, pakeisti esamų privilegijas, ir ar įmanoma išsiaiškinti esamų vartotojų duomenis – vartotojo vardą ar slaptažodį. Ištirtos tiek bandymas pakeisti vartotojų duomenis, susijusius su projektu, vietiniam kompiuteryje, tiek ir įklijuojant su projektu susijusių bylų, pakeistų kitame kompiuteryje, galimybės. Išsiaiškinta, kad turint sistemos administratoriaus teises, sistemą galima sutrikdyti visiem bandytiem posistemiam. Galimybė pridėti naujus vartotojus, ar keisti esamų slaptažodžius bei privilegijas yra InTouch ir CitectSCADA paketuose. Tuo tarpu WinCC pakete neįmanoma dešifruoti esamų vartotojo duomenų ar slaptažodžio, reikia kopijuoti visą projektą ir jį redaguoti.

Nagrinėjant saugumą OS lygmenyje, buvo tiriama, kokią įtaką turės esamų vartotojų grupių keitimas, vieno vartotojo priskyrimas kelioms grupėm bei naujo vartotojo sukūrimas – kada leidžiama prisijungti su juo. Atsižvelgiant į tai, kad vartotojų autentifikavimas nesiskiria lokaliame ir nuotoliniam lygmenyje, tirtas tik vienas iš šių – lokalus InTouch ir CitectSCADA bei nuotolinis su WinCC. Papildomai InTouch paketui ištirtas ir nuotolinio tipo ArchestrA saugumas. Išsiaiškinta, kad visiem paketam sukūrus naujus vartotojus, su jais galima iškarto prisijungti runtime aplinkoje. Pakeitus vartotojų grupes, priskiriamos privilegijos iškarto pasikeičia runtime aplinkoje, tereikia iš naujo prisijungti. Priskyrus vieną vartotoją skirtingom grupėm InTouch pakete, privilegijos priskiriamos priklausomai nuo scripto sudėties. CitectSCADA pakete, vartotojam priskiriamos visos privilegijos, būdingos kiekvienai grupei.

Siekiant ištirti nuotolinės bazės saugumo lygmenį, buvo tiriama ArchestrA IDE sistema. Joje sukonfigūruoti vartotojai, pasirinkus Galaxy tipo saugumą. Sukuriami keli vartotojai ArchestrA IDE aplinkoje, ir bandoma dešifruoti jų duomenis. Rezultatas – duomenys sėkmingai užšifruoti. Toliau sukuriama merged tipo aplikacija, įvedami keli objektai, parodantys prisijungimo lygį. Atsižvelgiant į tai, kad vartotojų duomenų bazės pakeitimus galima atlikti tik išjungus WindowMaker ir WindowViewer programas, vienintelis nagrinėtinas dalykas – privilegijų priskyrimas, vartotojui priklausant kelioms grupėms. Rezultatas – priskirta aukščiausias privilegijos lygis iš visų priskirtų vartotojo grupių.

Valdemaras Tubutis. Security analysis on supervisory control and data acquisition applications *Cotrol system Master's* thesis Assoc. Prof. Dr. Leonas Balaševičius. Kaunas University of Technology, Faculty of Electrical and Electronics Engineering, Department of Automatics.

Research area and field: Electrical and Electronics Engineering, Technological Sciences

Key words: SCADA, User, security

Kaunas, 2017. 44 p.

## SUMMARY

In this project user authorization methods and user access areas in supervisory control and data acquisition systems are reviewed. Three main software packages WinCC, InTouch and CitectSCADA are analysed. In addition, there is a review about possible attacks against SCADA system through network and explanation why most attention is given to that field. Considering that fact that the most attention is given to that field, a decision was made to analyze user authorization. The security of user authorization analysis is divided into two parts – application level, and operating system level. Not only that, but also the effect of deleting files associated with project or program executable files is checked. All research is made having only operating system administrative privileges.

Analyzing security in application layer attempts were made to make it clear if it is possible to add new users, change existing user roles, or to find out user name and password while having operating system administrator privileges. Both methods, when changes were made to user data in same computer, or when they were changed in other and then pasted were analyzed as well. It was found out that it is possible to crash the system, or make any futher work impossible when you have administrative privileges of the system. It is possible to add or change roles or passwords of existing users in InTouch or Citect packets, while WinCC has no such possibility – u need to copy or edit whole project.

Regarding user authorization under operating system security level changes were made to existing user groups, adding one user to several groups and creating one user – when it is possible to log in with it. Considering the fact that authorization methods are the same in local or remote modes, only of the following were analised for each – local for InTouch and CitectSCADA, and remote for WinCC. Additionally for InTouch packet ArchestrA security method was analyzed. It was found out that once you create new users in operating system, you can log in with them in all packets at runtime environment. After changing the user groups, privileges associated with users change automatically as well, all you need to do is log in again. If you add user to several groups, in InTouch they get privileges regarding how script is written, as for CitectSCADA, users are given privileges associated with each group.

As for remote database security analysis, ArchestrA IDE was selected. A few users were created, and then attempts to decrypt their data were made. Result – user data is successfully encrypted. Second step was to create a merged application, with a few objects, indicating current access level. After finding

out, that it is impossible to change user database in ArchestrA IDE while WindowViewer or WindowMaker is running, decision was made to test what access level user will get, when he belongs to several user groups. Result – the user was given highest access level from all groups he was appointed to.

# Turinys

# Įvadas

Supervizorinio valdymo ir duomenų surinkimo sistemos (toliau – SCADA) sistemos yra neatsiejama šių dienų infrastruktūros dalis. Aukščiausiame lygyje, SCADA sistemos valdo energetikos, vandens tiekimo ir transporto sistemas. Būtent šiose sistemose ypač aktualu tampa saugumas.

SCADA žmogus-mašina sąsaja (toliau HMI) yra vartotojo sąsaja su sistema, kurioje duomenys yra apdorojami ir atvaizduojami operatoriui. Sąsaja taip pat dažnai turi įgalintą sistemos valdymą. Taip pat leidžia lengviau stebėti įvairius RTU (remote terminal unit) ar PLC (Programmable logic controller). HMI paprastai yra susieta su sistemos duomenų bazėmis ir programine įranga tam, kad galėtų atvaizduoti grafikus bei įvairius duomenis, kaip suplanuotos aptarnavimo operacijos, logistikos informacija, detalias schemas. Svarbiausia informacijos dalis yra aliarmai. Aliarmai tai yra skaitmeninis statusas, kurio vertė gali būti norma arba aliarmas. Dažnai esant aliarmui, signalai ne tik atvaizduojami grafiškai, tačiau taip pat išsiunčiami elektroniniai laiškai ar tekstinės žinutės operatoriui.

Tinkamai neapsaugota SCADA HMI sistema gali suteikti visišką kontrolę įsilaužėliui. Ją gavęs, įsilaužėlis gali pavogti informaciją, sutrikdyti procesus, ar padavus klaidingas komandas netgi sugadinti įrangą ar turtą. Daugelis tyrimų yra atliktą analizuojant SCADA sistemos tinklo saugumą, tačiau aš nusprendžiau tirti būtent SCADA HMI programinės įrangos taikomųjų programų saugumą, kurio metu išsiaiškinsiu, ar įmanoma atlikti pakeitimus, leidžiančius bent dalinai perimti sistemos kontrolę.

*Darbo tikslas*: apžvelgti supervizorinio valdymo ir duomenų surinkimo taikomųjų programų apsaugos posistemių vartotojų saugumui taikomas priemones, jas ištirti, bei išsiaiškinti galimybes sutrikdyti sistemos darbą.

# 1. Apžvalginė dalis

## 1.1. Apsaugos priemonių SCADA HMI programose apžvalga

### 1.1.1. InTouch sistema

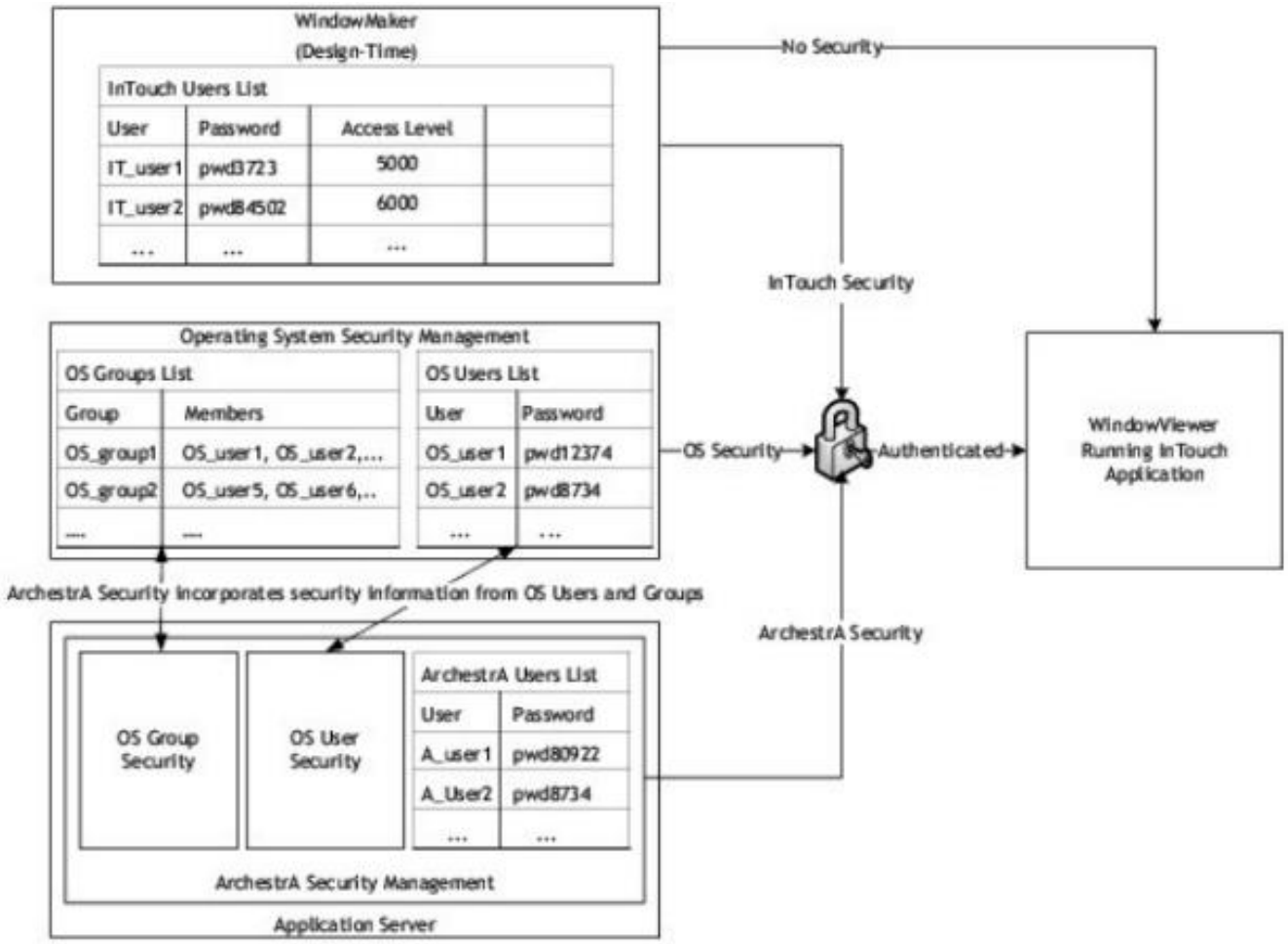
Šioje sistemoje yra taikoma trijų tipų apsaugos:

- Tradicinė, InTouch pagrindo apsauga (Traditional InTouch - based security);
- Operacinės sistemos pagrindo apsauga ( Operating system - based security);
- ArchestrA pagrindo apsauga (ArchestrA - based security).

Visos šios trys apsaugos atlieką tą pačią funkciją – draudžia neautorizuotą prisijungimą prie sistemos. Prisijungiant vartotojui sistema ne tik tikrina įvestus vartotojo duomenis, tačiau taip pat ir kokios privilegijos yra suteiktos vartotojui.

InTouch pagrindo apsauga patikrina, ar teisingi duomenys bandant prisijungti InTouch programoje. Kaskart, kai sukuriamas naujas projektas, būna tik vienas vartotojas – Administratorius, kurio slaptažodis wonderware. Visgi siekiant bent kiek apsaugoti sistemą, sukuriamas naujas vartotojas, su atitinkamu privilegijų lygiu, ir ištrinamas senas, administratoriaus teises turintis vartotojas.

Taikant operacinės sistemos apsaugą, dalis saugumo taisyklių yra paveldimos iš operacinės sistemos nuostatų, pvz. slaptažodžio gyvavimo trukmė. Šioje apsaugoje vartotojų vardai gali būti pasirenkami iš Windows Network Domain ar Workgroup sąrašų. Kadangi slaptažodžiai yra saugomi operacinės sistemos lygmeny, InTouch programoje jie nefigūruoja. Pasirinktiems vartotojams yra priskiriamos tik privilegijos. Taikant ši apsaugos būdą, vartotojams draudžiama keisti slaptažodį, atsijungti, prisijungi, ar pridėti naujus vartotojus.




**1.1 pav.** Taikytinos saugumo sistemos InTouch sistemoje. [1]

ArchestrA pagrindo apsauga leidžia prisijungti visiem iš Galaxy tinklo. Tam, kad būtų galima prisijungti nuotoliniu būdu taikant šį metodą, projekte turi būti specialiai nustatyta tokia galimybė, pasirenkant autentifikavimo metodą – Galaxy, OS User based, ar OS Group based. Visas saugumo konfigūravimas atliekamas IDE aplinkoje. Visgi funkcijos yra apribotos taip pat, kaip ir operacinės sistemos apsaugoje. Detalus ryšys tarp visų taikytinų saugumo sistemų InTouch aplinkoje pateiktas 1.1 pav. Būtina paminėti, kad norint prisijungti nuotoliniu būdu taikant ArchestrA metodą reikalinga papildoma programinė įranga.

Kaip papildoma apsauga gali būti naudojamos neveiklumo, meniu slėpimo ar klavišų ir jų kombinacijų naudojimo draudimas. Meniu slėpimas leidžia nerodyti tam tikros informacijos, ar uždrausti atlikti veiksmus pasirinktiems vartotojams.

Neveiklumo apsaugos tikslas yra tas, kad operatoriui palikus darbo vietą, niekas kitas negalėtų atlikti jokių veiksmų. Apsaugos veikimas paremtas tuo, kad jeigu vartotojas neatlieka veiksmų per tam tikrą laiko tarpą, sistema jį automatiškai atjungia. Dažnai yra panaudojami priminimai, kad būtina atlikti veiksmą, priešingu atveju būsite atjungtas.

Klavišų ir jų kombinacijų naudojimo draudimas neleidžia operatoriui įsijungti užduočių tvarkytuvo, ar pereiti į kitą programą. Tipiniai draudimai taikomi klavišų kombinacijoms kaip ctrl+alt+del arba alt+tab.

### 1.1.2. ArchestrA IDE sistema

ArchestrA apsauga apsaugo vartotojus nuo neleistinų veiksmų, įskaitant šiuos atvejus:

- IDE, kai konfigūruojami ir tvarkomi objektai;
- ArchestrA sistemos valdymo konsolė (System management console), kai atliekama priežiūros ir sistemos administravimo funkcijos;
- Vykdant visas kasdienines užduotis.

Apsauga ne tik valdo prieigą prie vartotojo sąsajos ArchestrA apsaugoje, bet taip pat ir prie objektų aprašymų ir duomenų. Kiekviena Galaktika Galaktikų saugykloje (Galaxy Repository) valdo savo paties apsaugos modelį. Apsaugos schema, valdoma Galaktikoje, yra trijų lygių konfigūravimo modelis, kurį sudaro sukūrimas ir priežiūra šių elementų:

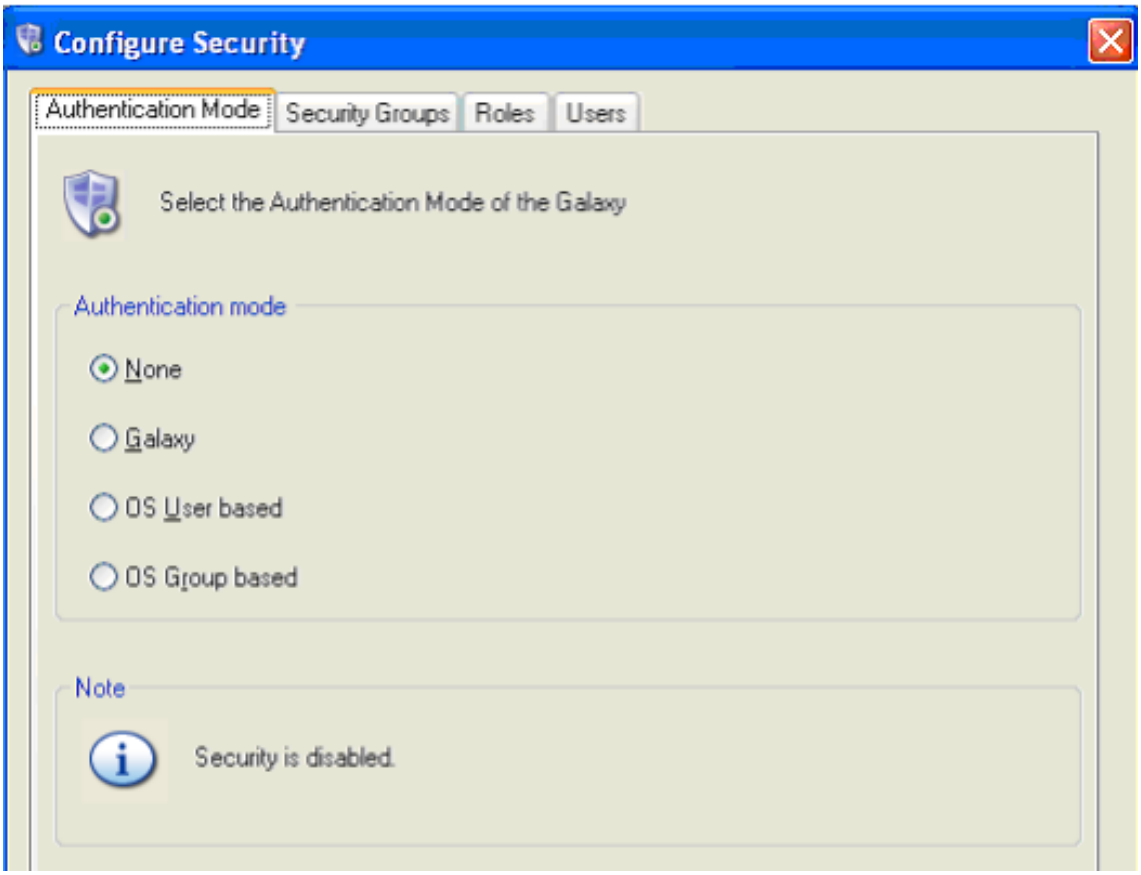
- Apsaugos grupės, susietos su specifiniais objektais Galaktikoje;
- Vartotojų pareigų, susietų su specifiniais administravimo, konfigūravimo ir valdymo leidimais, atsispindinčių apsaugos grupėse;
- Vartotojų, susietų su išskirtinėmis teisėmis.

Tokio tipo apsaugos matrica aprašo kaskadinį modelį vartotojų, susietų su specifinėmis rolėmis, kurios susietos su specifinėmis apsaugos grupėmis, kurios susietos su specifiniais objektais. Tokio tipo modelis leidžia varijuoti galimybėmis vartotojams nuo objekto iki objekto, nuo veiksmo iki veiksmo, nuo proceso iki proceso.

Tam, kad atsirastų prieiga prie Galaktikos apsaugos valdymo, yra trys kriterijai, būtini išpildyti:

- Joks kitas vartotojas negali būti prisijungęs prie Galaktikos;
- Visi objektai Galaktikoje turi būti patikrinti;
- Vartotojas, su kuriuo bandoma prisijungti prie Galaktikos apsaugos valdymo, turi turėti jam suteiktas teises, jei apsauga buvo prieš tai konfigūruota;

Jeigu bandoma prisijungti nesant nors vienam iš šių reikalavimų išpildytam, bus gauta perspėjimo žinutė ir prieiga bus uždrausta. Taip pat, prisijungus prie Galaktikos ir ją konfigūruojant, kitam vartotojui bandant prisijungti, prieiga bus uždrausta. Konfigūruojant apsaugą, pirmas žingsnis yra pasirinkti autentifikavimo metodą (1.2 pav.).




**1.2 pav.** ArchestrA IDE patvirtinimo apsaugos konfigūravimo langas [3].

Pasirinkus *None*, bus išlaikomas gamyklinis nustatymas naujoms Galaktikoms, kuris laikomas atviros apsaugos. Šis metodas leidžia visiems vartotojams atlikti visas funkcijas, nereikalaujant patvirtinti jų tapatybės net atliekant konfigūracinius pakeitimus.

Pasirinkus *Galaxy*, sistema naudojasi vietinės Galaktikos konfigūracija patvirtinant vartotoją. Šis metodas naudojamas siekiant pritaikyti vartotojų apsaugų sistemą, kontroliuojamą Galaktikos duomenų bazės.

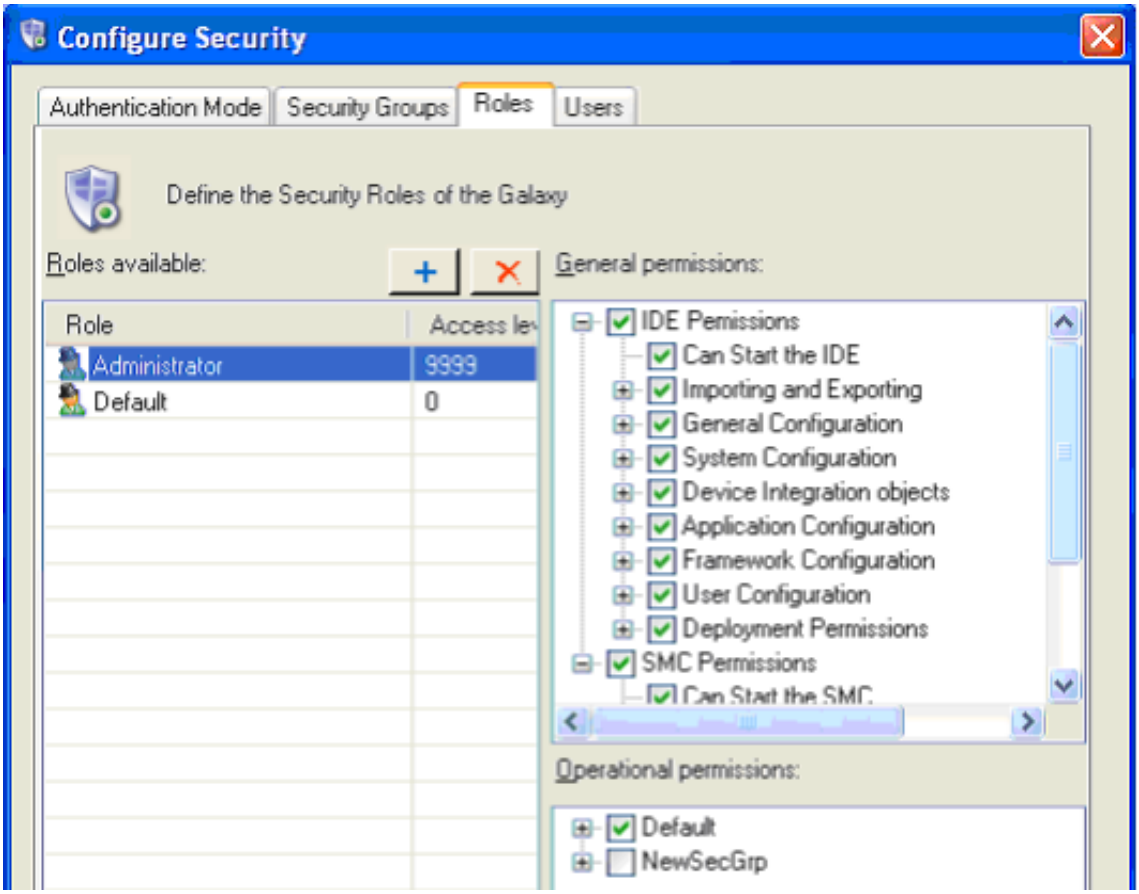
Pasirinkus *OS User Based*, įgalinamas patvirtinimas individualios operacinės sistemos vartotojų, leidžiant pilnai išnaudoti operacinės sistemos saugumą autentifikuojant vieną vartotoją.

Pasirinkus *OS Group Based*, įgalinamas patvirtinimas grupės vartotojų, kurie buvo priskirti operacinėje sistemoje. Pasirinkus šį metodą, galime įvesti papildomus parametrus, kaip prisijungimo laiko ar leistinų funkcijų atnaujinimo intervalų trukmę.

Sekantis žingsnis yra sukurti apsaugos grupes. Kiekvienas objektas Galaktikoje gali priklausyti tik vienai apsaugos grupei. Šios saugumo grupės yra atvaizduojamos rolėse. Žemiau pateiktame pavyzdyje (1.3 pav.), visi objektai yra priskirti *Default* apsaugos grupei.

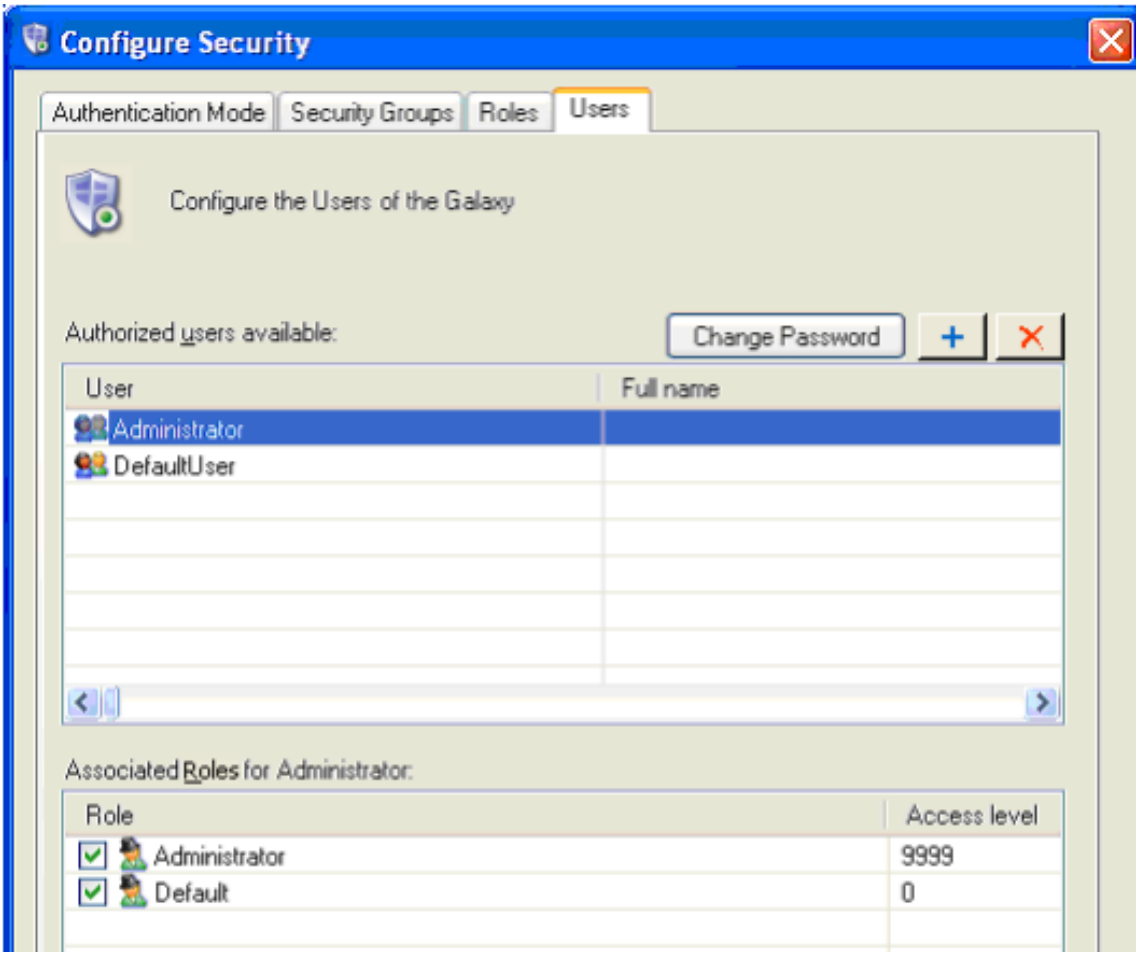

**1.3 pav.** ArchestrA IDE apsaugos grupių konfigūravimo langas [3].

Priešpaskutinis žingsnis yra rolių priskyrimas, kurios susijusios yra su programos konfigūravimu ir valdymu bei saugumo grupių paskyrimu. Pagal nutylėjimą Administratorius turi visas privilegijas, ir jo privilegijos negali būti modifikuojamos. Visgi sukūrus naują rolę, joje galima detaliai paskirstyti leidimus. Rolių gali būti neribotas kiekis. Vienintelis niuansas – rolių pavadinimai turi skirtis. Taip pat nurodomas kiekvienai rolei prieigos lygis, pagal kurį skirstosi prioritetai sistemos užduočių vykdyme. Jeigu vienam vartotojui priskirta daugiau, negu viena rolė, aukštesnio prieigos lygio rolė yra perduoda InTouch programai. Žemiau (1.4 pav.) pateiktas rolių redagavimo langas.




**1.4 pav.** ArchestrA IDE apsaugos rolių konfigūravimo langas [3].

Paskutinis žingsnis yra sukurti vartotojų sąrašą, pagal kurį bus galima prisijungti prie sistemos, nurodant kokios rolės priklausys kiekvienam vartotojui. Jei pasirinktas autentiškumo patvirtinimo metodas OS Group Based, tada šiame lange galima tik patikrinti informaciją, tačiau draudžiama pridėti naujus vartotojus, nes jie automatiškai papildo sąrašą kaskart prisijungus prie kompiuterio. 1.5 pav pateiktas vartotojų redagavimo lango pavyzdys.



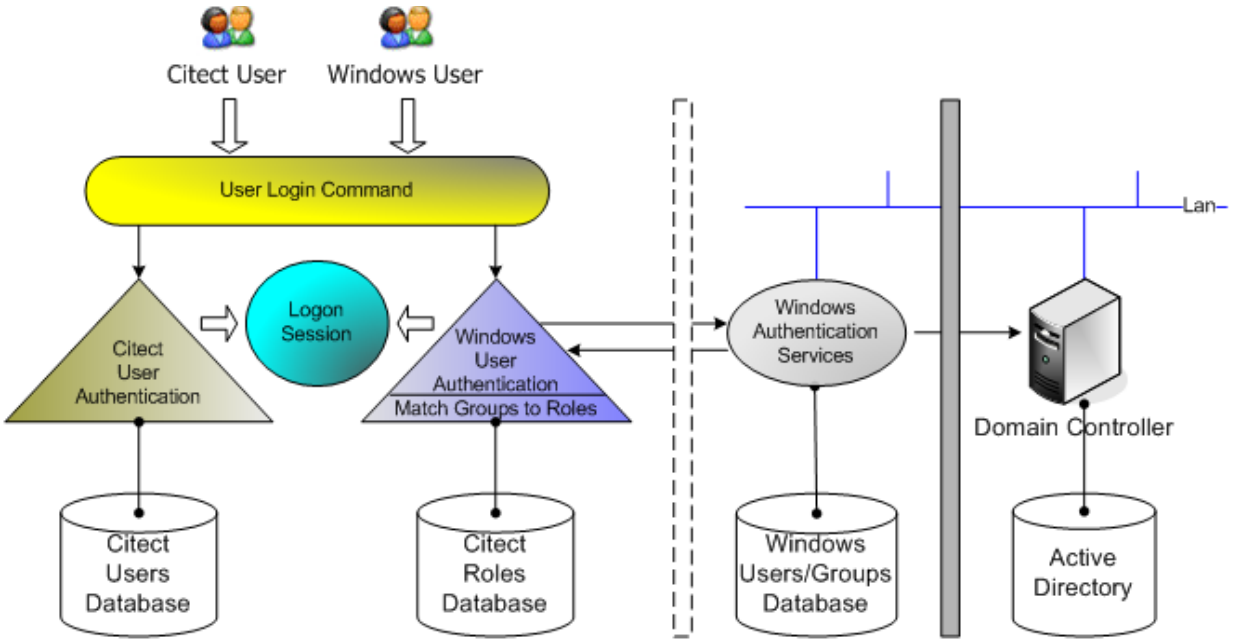

**1.5 pav.** ArchestrA IDE vartotojų konfigūravimo langas [3].

### 1.1.3. CitectSCADA sistema

Pati pagrindinė apsaugos priemonė visose SCADA sistemose yra vartotojo prisijungimo reikalavimas. Kiekvienas naujai sukurtas vartotojas privalo turėti vardą ir slaptažodį. Taipogi kuriant naujus vartotojus jiems priskiriama klasė – operatorius, vykdytojas, ar vadovas. Visgi visos klasės turi vieną bendrą požymį – privilegijos. Būtent privilegijų pagalba yra nurodoma, kokie veiksmai ir kokios informacijos priėjimas yra leistina kiekvienam vartotojui. Paminėtina, kad privilegijų sistema šioje programoje nėra hierarchinė, leidžiant suteikti visiškai skirtingas privilegijų grupes, pvz., 1 grupę, leidžiančią valdyti konvejerius ir 5, leidžiančią atspausdinti pranešimus.

Pakeitimai vartotojų duomenų bazėje, atlikti prisijungus prie sistemos realiu laiku, yra atvaizduojami tiktai vietiniuose _Users.rdb ir Users.dbf failuose. Tam, kad užtikrinti vartotojų duomenų sinchronizaciją, sistemos administratorius pakeitimus turėtų atlikti tiktai centriniame kompiuteryje. Jeigu projektas yra atstatomas iš kopijos (backup), tada visi vartotojų duomenys yra pašalinami ir atstatomi pradiniai. Tokiu atveju būtina imtis procedūrų pasinaudojant naujausiu Users.dbf failu, siekiant išvengti asinchronizacijos.

CitectSCADA suteikia galimybę susieti CitectSCADA vartotojus ir apsaugos pasirinktis kartu su standartine Windows OS apsaugos sistema, išlaikant tą pati CitectSCADA saugumo funkcionalumą, leidžiantį aprašyti vartotojus projekte ir prisijungti realiu laiku prie CitectSCADA. Taikant Windows OS apsaugos sistemą, Windows vartotojai gali prisijungti prie CitectSCADA sistemos realiu laiku, turėdami tam tikras privilegijas ir leistinas veikti sritis. Tam Windows vartotojui turi būti suteiktos atitinkamos rolės CitectSCADA sistemoje. Tam, kad suteikti Windows vartotojui roles, turi būti sukurta atskira rolė, kuri nurodo Windows vartotojų grupę, kuriai jis priklauso, ir pati SCADA sistema turi būti pridėta/priklausyti tai pačiai sričiai (domain). Taip pat yra numatyta automatinio prisijungimo funkcija *AutoLogin*, kai Windows vartotojas yra susietas su atitinkama role. Ši funkcija yra nustatoma keičiant *Citect.ini* faile AutoLoginMode nustatymą. CitectSCADA ir Windows OS apsaugų diagrama pateikta 1.6 pav.




**1.6 pav.** CitectSCADA apsaugos ir Windows OS apsaugų diagrama. [2]

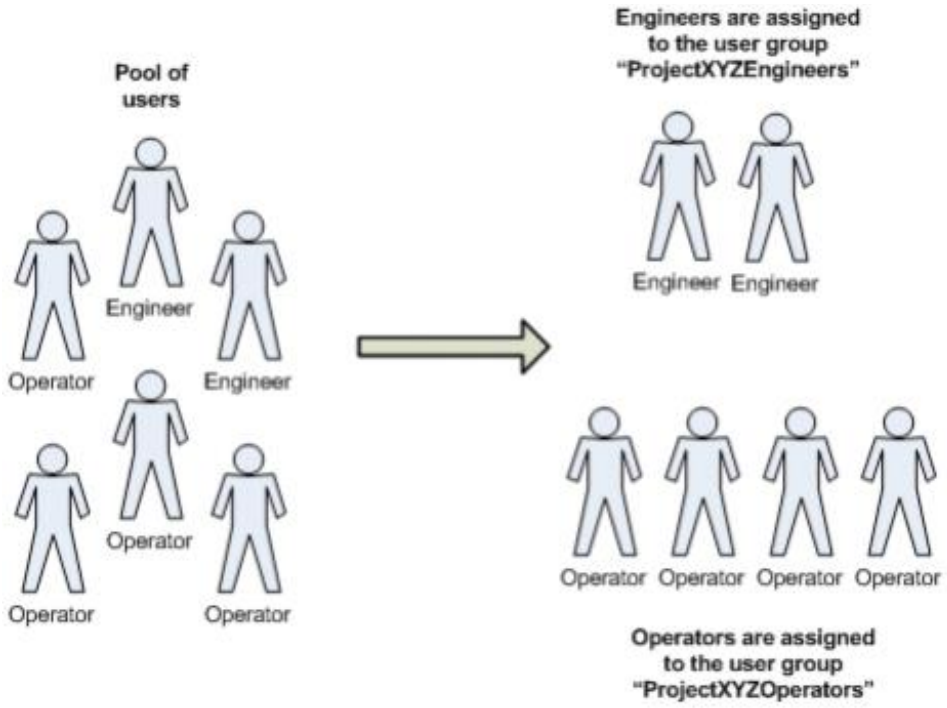
Prisijungiant prie sistemos Windows apsaugos metodu, galimi šie prisijungimo išlygų atvejai:

1. Vietinis vartotojo prisijungimas. Kai patvirtinamas Windows vartotojas, nesant rolių/grupių vardams susietiems su tam tikra sritimi (domain), tada atsitiktinė tinklo sritis bus naudojama patvirtinti vartotojo prisijungimui.
2. Srities prisijungimas. Esant rolių/grupių vardams susieti su sritimi, Windows bandys patvirtinti srities vartotojus ir grupes. Jei srities valdikliai yra neprieinami, tada naudojama informacija iš tarpinės atminties (cached).

3. Nuotolinio kliento prisijungimas. Kai atliekamas nuotolinis prisijungimas iš srities ar patikimos srities (trusted domain), pati CitectSCADA sistema nevykdo vartotojo autentiškumo patvirtinimo, o tik patvirtina kada vartotojas atsijungia.

Tam, kad iš tinklo kompiuterio būtų galima nevaržomai prisijungti prie CitectSCADA sistemos, jis turi priklausyti patikimam tinklui. Yra sukuriamas slaptas mašininis kodas, pagal kurį kompiuteriai vienas kita autentifikuoja. Slaptažodis yra sukonfigūruojamas per *Computer Setup Wizard*, ir nustatomas *Citect.ini* faile. Jeigu slaptažodis nebuvo tinkamai sukonfigūruotas, tačiau jo reikalauja *Citect.ini* failo nustatymai, serveris tampa nebetinkamas veikti ir atitinkama žinutė yra suformuojama sistemos stebėsenoje *syslog*.

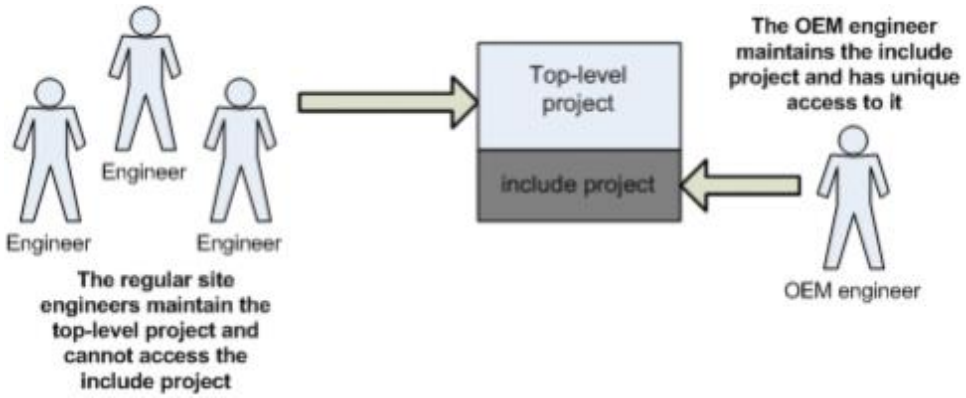
Kaip papildomas apsaugos priemones, šiame pakete gamintojai yra įdiegę galimybę panaikinti sistemos kontrolės meniu ir sistemos pasileidimo atšaukimo mygtuką, bei klavišų nuoseklaus paspaudimo siekiant įvesti komandas (key sequences for commands).




**1.7 pav.** Vartotojų padalinimas į dvi atskiras grupes Citect HMI aplinkoje. [2]

Siekiant išvengti projekto pakeitimų, jo privilegijų tipas yra pakeičiamas į *Read-only*, kuris draudžia keisti privilegijas esamo vartotojo projekte. Visgi anksčiau ar vėliau gali prireikti ir jį koreguoti, dėl to yra priskiriama, kuriems vartotojams suteikti galimybę koreguoti projektą, kuriems ne. Pavyzdys pateiktas 1.7 pav., kur sukurtos dvi vartotojų grupės – inžinieriai, atsakingu už projekto konfigūraciją ir operatorių, atsakingų už operacijų vykdymą. Taip pat yra galimybė sudaryti įterptinį projektą, kuriame tik gamintojas gali atlikti pakeitimų veiksmus, o inžinieriai jame turi tik *Read-only* privilegijas, o tuo

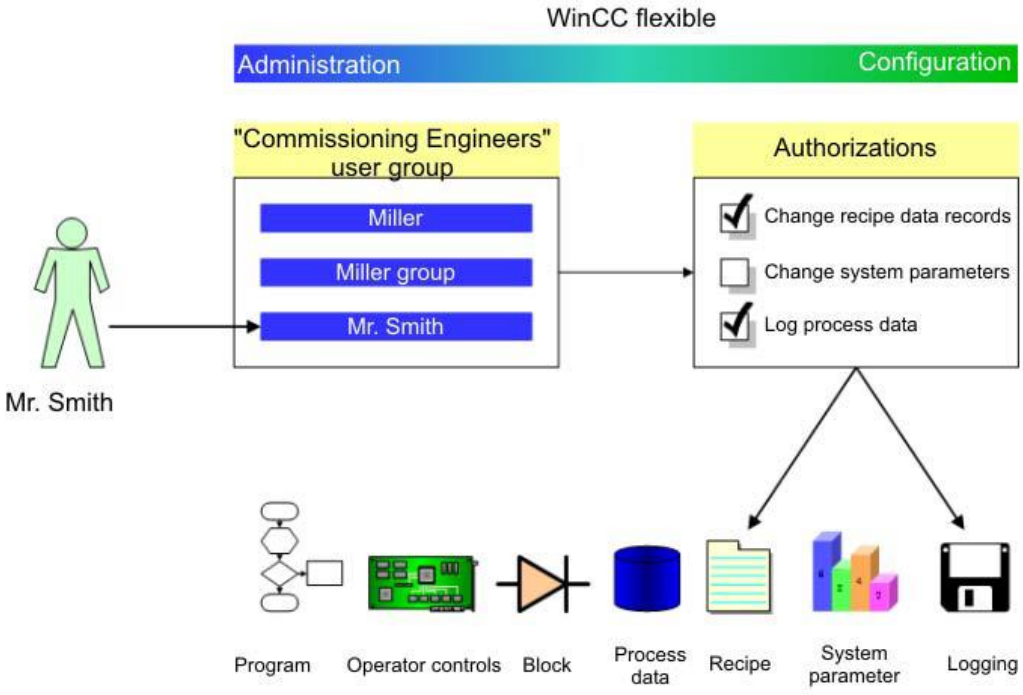
tarpu pagrindiniame projekte inžinieriai turi teisę jį koreguoti, o gamintojas tik *Read-only*. Toks pavyzdys pateiktas 1.8 pav.




**1.8 pav.** Įterptinių projektų sudarymas Citect HMI aplinkoje. [2]

### 1.1.4. WinCC sistema

Esant dideliam projektui, tampa pravartu išskirstyti privilegijas skirtingiems vartotojams. Privilegijos nebūtinai turi būti paskirtos individualiai kiekvienam vartotojui, dažnai užtenka priskirti grupėms. Visgi esant reikalui, sistema suteikia galimybę net ir individualiai paskirti privilegijas, taip suteikiant didelį lankstumą konfigūruojant vartotojus. Principinė schema, kurioje pateikta vartotojų konfigūracija pateikta 1.9 pav. Tačiau visa likusi apsauga, taikoma tokiu pat metodu, kaip ir prieš tai apžvelgti HMI.

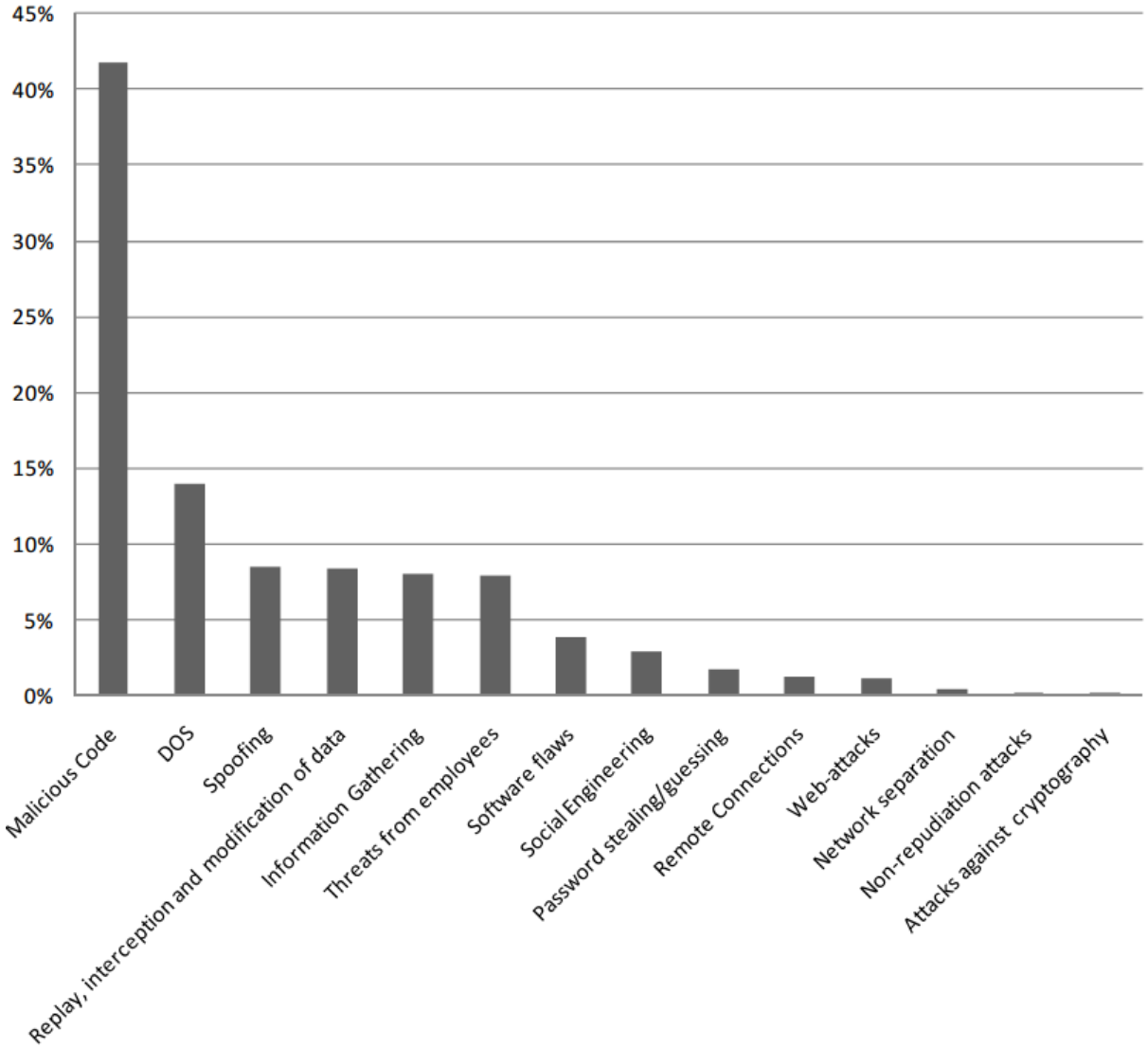



**1.9 pav.** Principinė vartotojų konfigūravimo schema. [4]

# 2. Statistika ir saugumo standartai

Statistikos, kurioje būtų pateikti įsibrovimo į sistemas ir žalos padarymo atvejai bei būdai nėra, nes tokios informacijos viešinimas keltų pavojų įmonių reputacijai [11].

Atsižvelgiant į SCADA sistemų saugumo aktualumą, sukurti daugybė saugumo standartų, kurie apžvelgia skirtingas kategorijas, ir yra išleisti daugybės institucijų - American Gas Association (AGA), National Institute of Standards and Technology (NIST), Centre for the Protection of National Infrastructure (CPNI), International Electrotechnical Commission (IEC), the North American Electric Reliability Corporation (NERC), IEEE. Visų šių standartų apžvalga atlikta straipsnyje „SCADA System Cyber Security – A comparison of Standards“[9], kuriame išnagrinėjama kiekvienos grėsmės grupės paminėjimas ir pateikiama diagrama 2.1 pav.




**2.1 pav.** Grėsmių paminėjimas standartuose [9].

# 3. Metodinė dalis

Tiriant SCADA saugumą, galima išskirti dvi sritis:

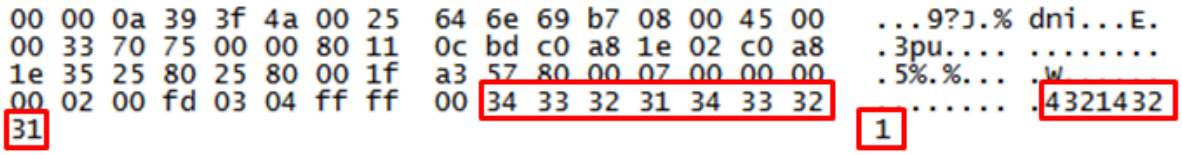
1) Tinklo lygmuo;
2) Aplikacijos lygmuo;

## 3.1. Tinklo lygmuo

Šiuo metu tinklo lygmeniu tyrimas yra plačiausiai atliekamas, kadangi įsilaužėlis gali prisijungti prie tinklo iš daugybės vietų. Vienas iš galimų testų tinklo lygmenyje yra atakų surengimas prieš PLC ar HMI serverį. Toliau bus apžvelgiamos atakos PLC lygmenyje.

*Kriptografinės atakos*. Šių atakų esmė yra išgauti duomenis nuskaitant paketus, siunčiamus tarp PLC ir valdomojo kompiuteriu. Šiai atakai surengti užtenka paprasto paketų skaitymo įrankio, kaip Wireshark. Sėkmingai nuskaičius paketus, galima sužinoti įvairius slaptažodžius:

1) skaitymo apsaugos slaptažodį (read protection password) 3.1 pav.;




**3.1 pav.** Read slaptažodis nuskaičius paketą wireshark programa [5].

2) FTP prieigos slaptažodį (FTP Access password), leidžiantį prieiti prie duomenų, įrašytų PLC atmintyje;
3) HTTP prieigos slaptažodį (HTTP Access password), suteikiantį beveik pilną PLC valdymą iš tinklo;
4) Įteisintų mazgų skaičių.

*Pakartotinės atakos*. Kaip viena iš apsaugų yra įrašymo apsauga (write protection), leidžianti priimti komandas tik iš atitinkamo šaltinio, su specialiu kodu. Siekiant jį išsiaiškinti, galima tiesiog siųsti rašymo komandas (write requests) visomis galimomis kodų konfigūracijomis ir stebėti, kada PLC atsakys į komandą (FINS protokole yra galimi 255 šaltinio numeriai su 127 galimais adresais, sudarant 32385 galimas kodo reikšmes ( 255x127=32385). Juos išsiaiškinus galima nusiųsti bet kokią komandą į valdiklį [5].

*Fragmentacinės atakos*. Šių atakų esmė yra siųsti tokius paketus, kurie yra netaisyklingos struktūros – ar per dideli, ar per trumpi, priverčiant juos priimantį įrenginį juos bandyti sudėlioti į tinkamą struktūrą [6]. Ši ataka identiška visiems įrenginiams. Jų galimos kelios variacijos:

1) Ping of Death – ataka atliekama siunčiant IP *ping* paketus didesnio dydžio, nei leidžia IPv4 standartas;
2) TearDrop – ataka atliekama siunčiant paketus su netaisyklingais padalinimo kodais, priverčiant paketus persidengti vienas su kitu ir taip perkrauti sistemą;

*DoS (Denial of Service) atakos*. Panašiai, kaip ir fragmentacinės atakos, DoS atakomis siekiama sutrikdyti įrenginio darbą siunčiant didžiulius kiekius paketų per trumpą laiko tarpą. Ši ataka identiška visiems įrenginiams. Tai galima atlikti šiais keturiais metodais:

1) UPD atsako ataka (UDP reflect ataka) – ataka atliekama siunčiant daugybę nuskaitymo užklausų. Jei įrenginys neturi įdiegto filtravimo pagal IP/MAC adresą ar mazgo numerį, jis atsakinės į kiekvieną užklausą.
2) Naudojant Netwox paketą – ataka atliekama pasinaudojus vienu iš daugelio įrankių, skirtų testuoti tinklą. Pavyzdžiui naudojant 76 įrankį, į PLC yra pasiunčiami didžiuliai kiekiai SYN tipo paketų [7];
3) WWW neribotų užklausų ataka (*WWW infinite attack request*) – ataka, kai kreipiamasi į PLC per HTTP prievadą (port);
4) *LOIC* (Low orbit ion canon) – įrankis, leidžiantis įvykdyti ataką siunčiant UPT/TCP/HTTP paketus labai dideliais kiekiais [8];

Atakos HMI lygmenyje.

*Kriptografinės atakos*. Šias atakas vykdyti naudojamas taip pat Wireshark įrenginys, galintis nuskaityti paketo duomenis, kuris išsiunčiamas vartotojui įvedus vartotojo vardą ir slaptažodį.

*Pakartotinės atakos*. Šios atakos principas yra analogiškas PLC dalyje apžvelgtam metodui. Bandoma atspėti būtinus specialiuosius šaltinio kodus, kuriuos atspėjus HMI reaguoja į bet kokias komandas – perkrovimą, išjungimą ir t.t. Būtina paminėti, jog ši funkcija veikia tik tuo atveju, jei HMI yra valdomas iš kito kompiuteriu (controller PC).

## 3.2. Aplikacijos lygmuo

Aplikacijos lygmenyje pagrindinis dėmesys skiriamas vartotojų autentifikavimui, ir pagal tipą grupuojamas į lokalų ir nuotolinį.

*Aplikacijos lygmuo*. Šiame lygmenyje tiriama keletas saugumo metodų ir sričių, pagrindinė – vartotojų saugumas. Tiriant aplikacijos lygmeny priimama, kad vartotojas turi sistemos

administratoriaus priėjimo lygmenį, tačiau neturi aplikacijos priėjimo lygmens. Keletas vartotojų saugumo metodų:

1) Radimas bylų, kuriose saugomi vartotojų duomenys, ir bandymas juos dešifruoti;
2) Ištrinti su vartotojo duomenimis susijusias bylas, siekiant sutrikdyti sistemos darbą;
3) Nukopijavimas esamo projekto, ir jame bandoma atlikti vartotojų duomenų pakeitimus. Tai atlikus bandoma pakeisti esamas su vartotojų informacija susijusias bylas veikiančiame projekte. Pakeitimai: administratoriaus slaptažodžio keitimas ir naujo vartotojo sukūrimas.

Tiriama, kokią įtaką turi bylų, susijusių su projektu ištrynimas ir pačia programine įranga ir galimybės apeiti taikomus vartotojų funkcijų apribojimus.

*Operacinės sistemos lokalus lygmuo.* Tiriant OS lokaliu lygmeniu, bandoma išsiaiškinti, kokią įtaką sistemos saugumui turi OS vartotojų keitimas – naujų vartotojų pridėjimas, esamų ištrynimas.

*Operacinės sistemos nuotolinis lygmuo.* OS nuotolinio lygmens tyrimas yra identiškas lokalaus lygmens tyrimui.

*Nuotolinės duomenų bazės lygmuo.* Atliekant nuotolinės duomenų bazės lygmens tyrimą, siekiama išsiaiškinti, kur toje bazėje yra saugomi duomenys, ir kokia įtaka saugumui keičiant juos.

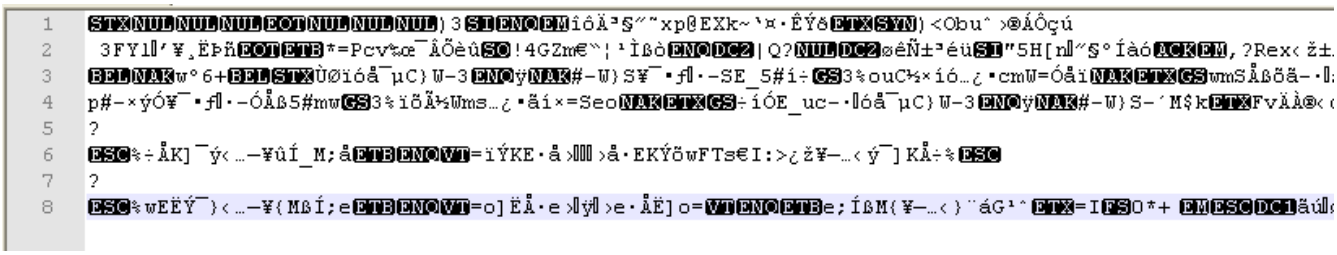
# 4. Tiriamoji dalis

Tyrimas atliekamas naudojant Oracle VM VirtualBox programinę įrangą, kuri sukuria virtualų kompiuterį. Jame įrašoma Windows XP SP3 32 bit operacinė sistema. Siekiant geriau ištirti aplikacijų saugumą, sukuriama antras toks pat virtualus kompiuteris.

## 4.1. Aplikacijos lygmuo

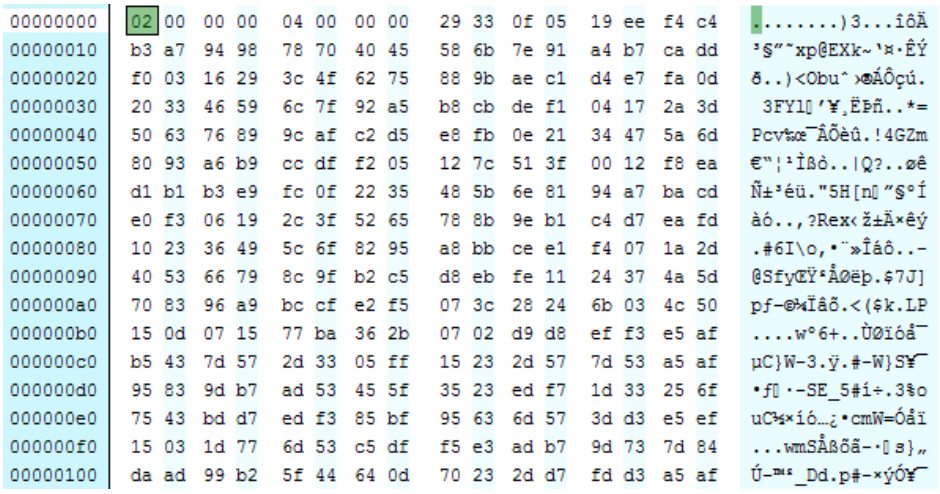
### 4.1.1. InTouch sistema

Remiantis vartotojo instrukcija [12], priimama, kad vartotojų duomenys ir slaptažodžiai saugomi *password.bin* byloje. Atvėrus bylą su notepad++ programa, jokių duomenų nepavyko nuskaityti (pavyzdys pateiktas 4.1 pav.).



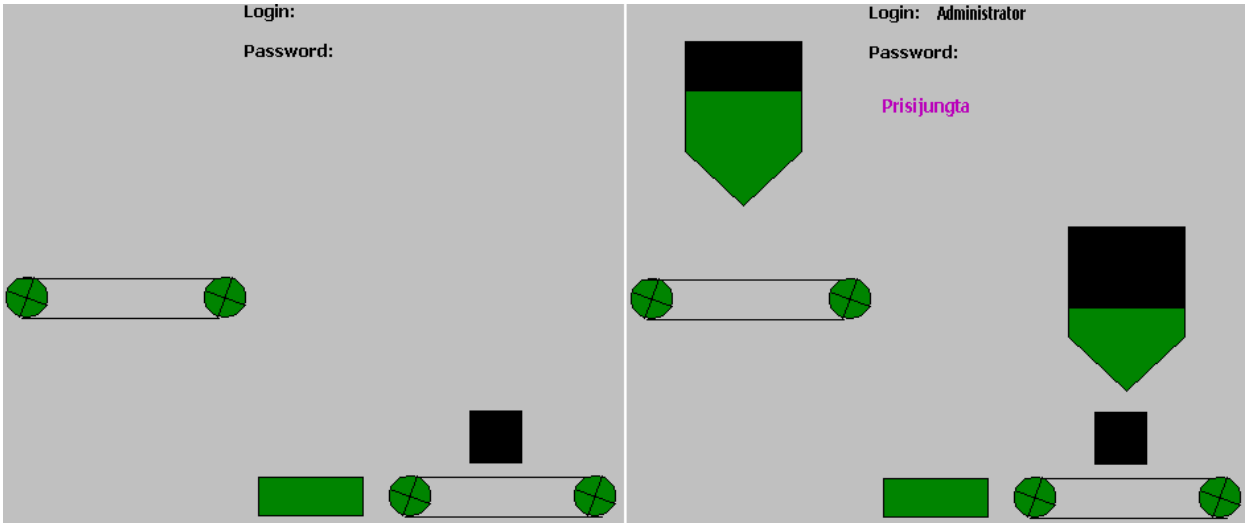
**4.1 pav.** password.bin byla atidarius su notepad++.

Turint omenyje, kad .bin bylos plėtinys reiškia, kad byla yra *binary* tipo, bandoma dešifruoti duomenis su hex editor programa [13]. Rezultatas (4.2 pav.) toks pat – duomenys sėkmingai užšifruoti ir jų neįmanoma perskaityti.



**4.2 pav.** Password.bin byla atidarius su Hex editor neo.

Sekantis žingsnis – projekto kopijavimas, vartotojų duomenų keitimas ir veikiančio projekto passwords.bin bylos pakeitimas. Kadangi administratorius turi galimybę keisti saugumo parametrus tiek *development*, tiek *runtime* aplinkoje, papildomai yra sukuriamas bandomasis projektas, kuriame įvedus teisingus vartotojo duomenis, atsiranda užrašas „Prisijungta“, ir atsiranda dvi talpos. Kadangi pradinio vartotojo „Administrator“ ištrinti neįmanoma net sukūrus naują vartotoją, yra pakeičiamas tik šio vartotojo slaptažodis iš wonderware į „1“. Bandomojo projekto pavyzdys pateiktas 4.3 pav., kur kairėje – vaizdas, matomas neprisijungus vartotojui, ir dešinėje, kai vartotojas jau prisijungęs.




**4.3 pav.** Bandomasis projektas.

Bandomasis projektas sukurtas pavadinimu „Atsiskaitymas“, ir išsaugotas į C:\Documents and Settings\Administrator\My Documents\My InTouch Applications. Projektas yra nukopijuojamas, ir pervadinamas į atsiskaitymas(2). Priima, kad nežinomas esamo vartotojo slaptažodis. Paleidus nukopijuotą projektą, neįmanoma prisijungti ar pakeisti vartotojų duomenų. Yra galimybė ištrinti password.bin bylą, prarandant visus vartotojų duomenis, tačiau atstatant gamyklinį aplikacijos administratoriaus slaptažodį – wonderware [14]. Projektas yra paleidžiamas per InTouch programą, pakeičiamas aplikacijos Administrator vartotojo slaptažodis į 2, sukuriamas naujas vartotojas su vardu „As“ ir slaptažodžiu „a“. Tada išbandomi šie penki variantai:

1) Password.bin ištrynimas, kai programa veikia realiu laiku;
2) Pakeisto password.bin įklijavimas, kai programa veikia realiu laiku;
3) Pakeisto password.bin įklijavimas, kai programa išjungta;
4) Kitame kompiuteryje pakeisto password.bin įklijavimas, kai programa veikia realiu laiku;
5) Kitame kompiuteryje pakeisto password.bin įklijavimas, kai programa išjungta.

*Pirmasis atvejis*. Ištrynus bylą, development aplinkoje **prisijungta** su wonderware, tuo tarpu runtime aplinkoje išlieka nepakitęs. Perkrovus projektą, slaptažodis pasikeičia į wondeware. Perkrovus programą, abiejose aplinkose prisijungta wonderware slaptažodžiu.

Antruoju variantu pakeitus password.bin bylą, perkrauti projekto net nereikėjo – prisijungimas su nauju slaptažodžiu buvo **sėkmingas** abejose aplinkose tiek kaip aplikacijos administratoriumi, tiek nauju vartotoju. Perkovus projektą/ programą, taip pat sėkmingai prisijungta tiek nauju aplikacijos administratoriaus slaptažodžiu, tiek nauju vartotoju.

Visais likusiais atvejais, kaip ir antruoju, pakeitus password.bin bylą, perkrauti projekto net nereikėjo – prisijungimas su nauju slaptažodžiu buvo **sėkmingas** abejais vartotojais abejose aplinkose. Perkovus projektą/ programą, prisijungta nauju vartotoju ir nauju administratoriaus slaptažodžiu.

Taip pat išbandoma galimybė įklijuoti password.bin bylą iš visiškai kito projekto, su naujais vartotojais. Rezultatas – **sėkmingai** prisijungta su naujais vartotojais ir slaptažodžiais.

Siekiant išsiaiškinti bylų ištrynimo įtaką, tiriama, ar sutriks sistemos darbas jas ištrynus veikimo metu, ir ar leis paleisti projektą iš naujo.

Ištrynus su projektu susijusius duomenis, sistemos veikimas nesutriko, prisijungta buvo senu slaptažodžiu. Paleidžiant projektą pradėjo mėtyti sistemines klaidas, projektas nepasileido. Galima teigti, jog sistemos darbas buvo sėkmingai **sutrikdytas**. Sisteminės klaidos pavyzdys pateiktas 4.4 pav.


**4.4 pav.** Sisteminė klaida, ištrynus projekto duomenis.

Sekantis žingsnis trinti programos sistemines bylas veikiant projektui. Veikimo metu pasirinkus aplanką InTouch, išmetė klaidą, pateiktą 4.5 pav.


**4.5 pav.** Sisteminė klaida, bandant trinti programos bylas.

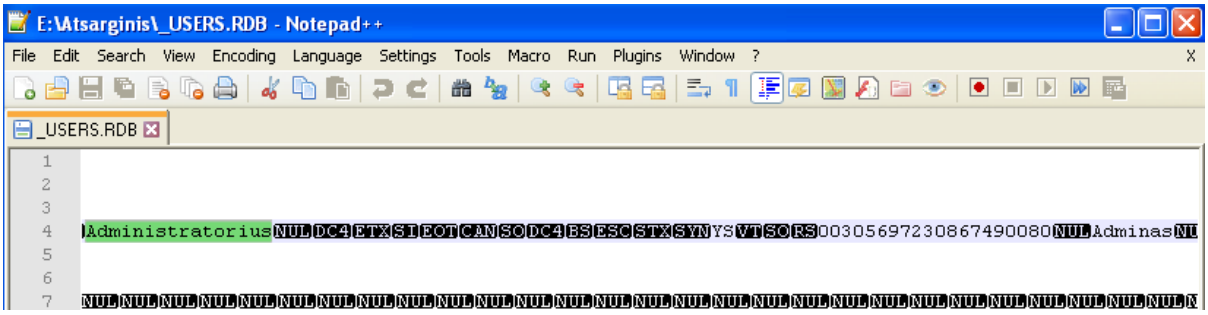
Visgi įėjus į aplanką ir pasirinkus po vieną bylą, sėkmingai buvo ištrintos 122 bylos. Sistemos veikimas nesutriko, tačiau perkrovus projektą, programa nustojo veikti, išmesdama sisteminės klaidos pranešimą, pateiktą 4.6 pav.


**4.6 pav**. Programos darbo sutrikimo langas.

Nepaisant ištrintų bylų kiekio, programą pavyko iš naujo paleisti, visgi paleisti projektą iš naujo, ar kitą – nepavyko, programa nustojo veikti ir išmetė tokią pat lentelę, kaip pateikta 4.6 pav. Pabandžius ištrinti bylas su LockHunter programa, programa išmetė tokią pat lentelę, kaip ir 4.6 pav., tačiau bylų, kartu ir InTouch aplankalo neliko. Programa sėkmingai ištrinta, liko tik projekto duomenys. Sistemos darbas **sėkmingai** sutrikdytas.

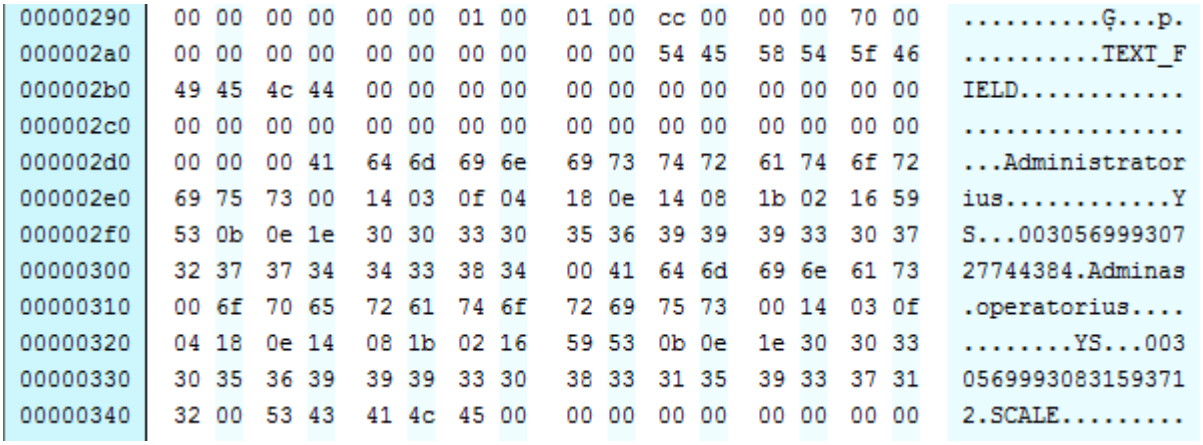
### 4.1.2. CitectSCADA sistema

Remiantis vartotojo vadovu [2], priimama, kad vartotojų duomenys ir slaptažodžiai saugomi _Users.rdb ir Users.dbf bylose. Sukuriamas projektas su dviem vartotojais – Administratorius, su slaptažodžiu slaptazodis, ir Operatorius, su slaptažodžiu „slaptažodis“. Atvėrus bylą su notepad++ programa, pavyko nuskaityti esamų vartotojų vardus ir priskirtas roles (pavyzdys pateiktas 4.7 pav.), tačiau slaptažodžiai – sėkmingai užšifruoti.




**4.7 pav**. _Users.rdb bylos duomenys nuskaityti su notepad++.

Atvėrus users.dbf bylą su notepad++, nuskaityti pavyko tokius pat duomenis, tačiau slaptažodžio – ne. Toliau tiriama su hex editoriumi. Bandant nuskaityti duomenis su hex editoriumi, rezultatai tokie apt, kaip su notepad – abiejose bylose nuskaityti tik vartotojų vardai ir priskirtos rolės. Pavyzdys pateiktas 4.8 pav.



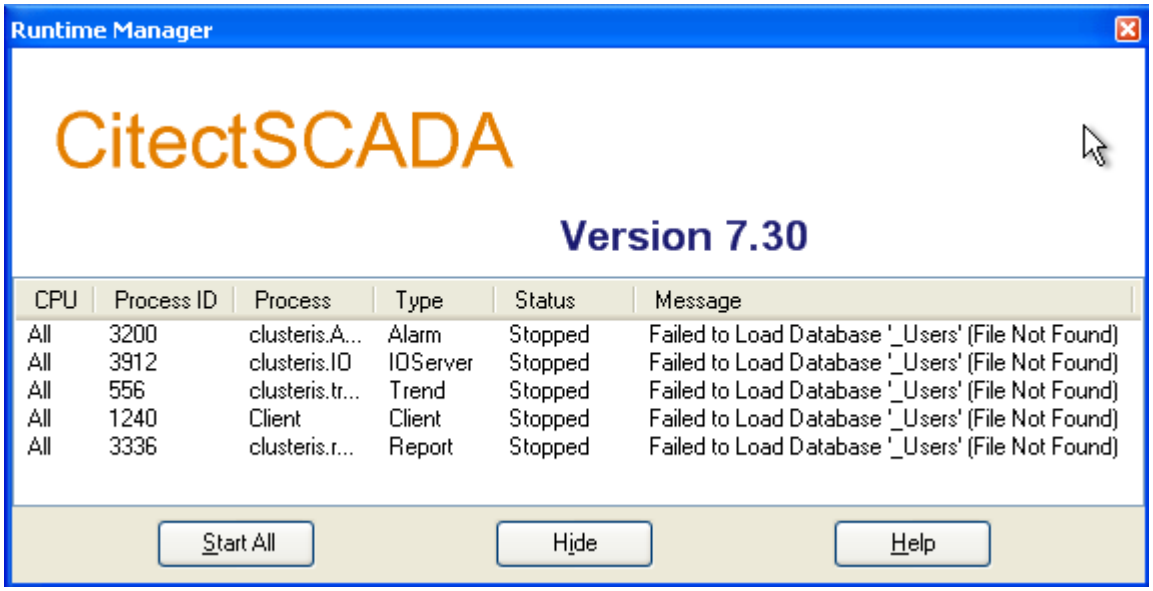

**4.8 pav**. _Users.rdb bylos duomenys nuskaityti su hex editor.

Siekiant ištirti vartotojų autentifikavimo apsaugą, tiriami šie septyni variantai:

1) _Users.rdb ir Users.dbf ištrynimas, kai programa veikia realiu laiku;
2) Pakeistų _Users.rdb ir Users.dbf įklijavimas, kai programa veikia realiu laiku;
3) Pakeistų _Users.rdb ir Users.dbf įklijavimas, kai programa išjungta;
4) Kitame kompiuteryje pakeistų _Users.rdb ir Users.dbf įklijavimas, kai programa veikia realiu laiku;
5) Kitame kompiuteryje pakeistų _Users.rdb ir Users.dbf įklijavimas, kai programa išjungta;
6) Kitame kompiuteryje sukurto atskiro projekto _Users.rdb ir Users.dbf įklijavimas, kai programa įjungta;
7) Kitame kompiuteryje sukurto atskiro projekto _Users.rdb ir Users.dbf įklijavimas, kai programa išjungta.

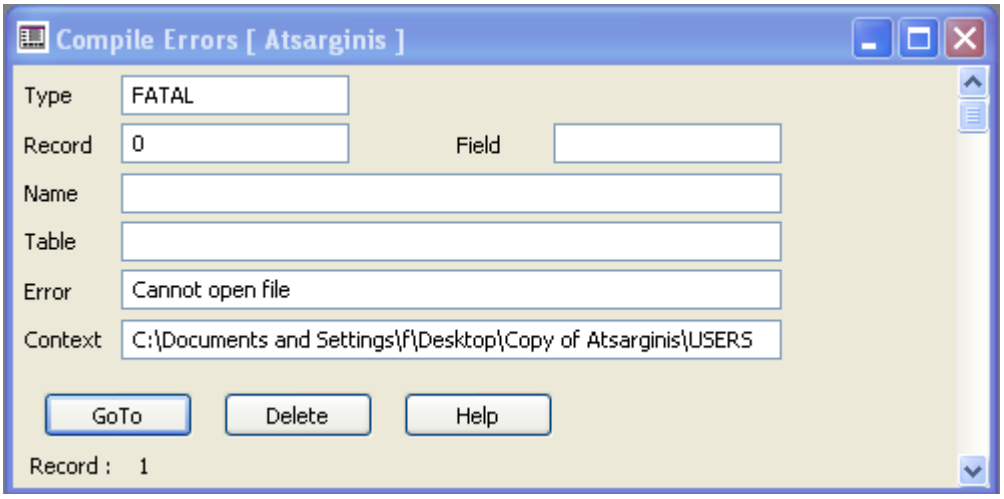
Kadangi tiriama ne tik galimybė prisijungti su naujais vartotojais, tačiau ir naujų vartotojų privilegijų praplėtimas, projekte numatyta galimybė patvirtinti aliarmus tik vartotojams su 1 privilegijos grupe.

*Pirmas variantas*. Ištrynus _users,rdb bylą, projekte sėkmingai buvo galima atsijungti ir prisijungti su esamais vartotojais. Perkrovus projektą, _users.rdb byla buvo automatiškai sukurta iš naujo. Ištrynus users.dbf bylą, prisijungimas su esamais vartotojais sėkmingas tiek veikiant projektui, tiek perkrovus programą. Visgi nauja users.dbf byla projekto aplankale neatsirado. Ištrynus abi bylas projekto veikimo metu, toliau sėkmingai prisijungta ir atsijungta su esamais vartotojais. Visgi išjungus projektą ir bandant įjungti, nepasileido serveriai ir atsirado gedimas teigiantis, jog nerasta duomenų bazė, susijusi su vartotojais (pavyzdys pateiktas 4.9 pav.).



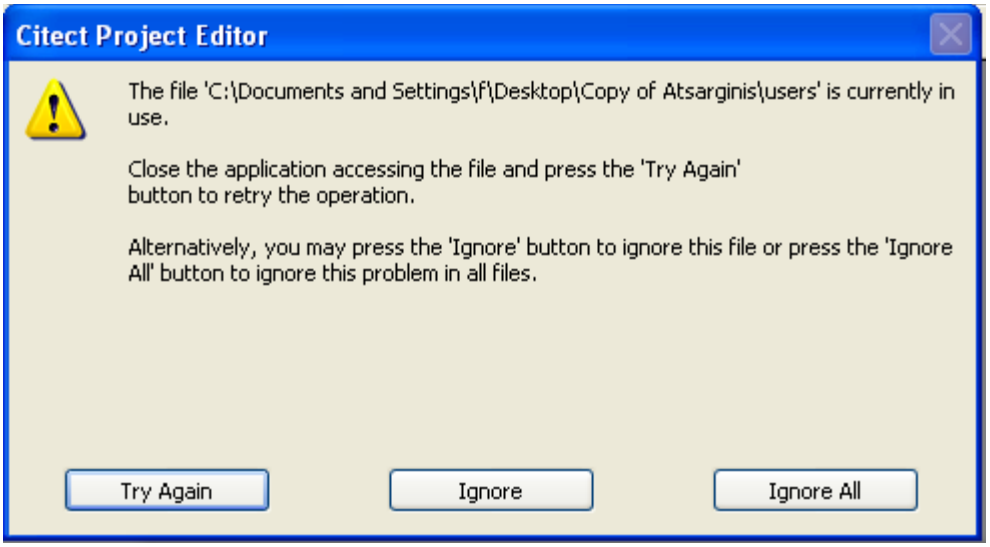

**4.9 pav**. Sistemos sutrikimo langas, kai ištrintos _Users.rdb ir Users.dbf bylos.

Tuo tarpu bandant sukompiliuoti, išmeta „fatal“ tipo gedimą, kurio neina išspręsti spaudžiant klavišą „Go To“ (pavyzdys pateiktas 4.10pav.).




**4.10 pav**. Projekto kompiliavimo klaida, kai ištrintos _Users.rdb ir Users.dbf bylos.

Bandant sukurti naują vartotoją, programa išmeta gedima, pateiktą 4.11 pav. Kadangi sukurti naujų vartotojų neina, galima teigti, jog sistemos darbas sutriko.




**4.11 pav**. Sisteminė klaida, kai CitectSCADA explorer aplinkoje bandoma sukurti naujus vartotojus.

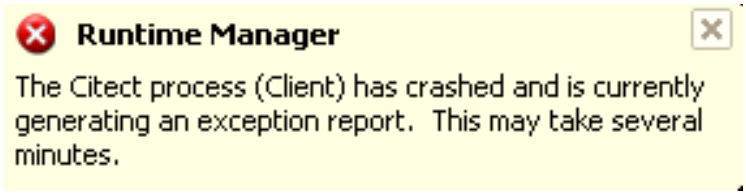
*Antras variantas*. Įklijavus pakeistas bylas, prisijungti su naujais vartotojais ir esamų pakeistais slaptažodžiai nepavyko, nei perkrovus projektą, nei perkrovus programą. Tik iš naujo sukompiliavus projektą buvo sėkmingai prisijungta su naujais vartotojais, patvirtintas aliarmas.

*Trečias variantas*. Įklijavus pakeistas bylas, kai programa išjungta prisijungti su naujais vartotojais ir slaptažodžiais nepavyko, galiojo seni. Visgi perkompiliavus projektą, pradeda galioti nauji vartotojai ir slaptažodžiai.

*Ketvirtas variantas*. Įklijavus pakeistas bylas, prisijungti su naujais vartotojais ir esamų pakeistais slaptažodžiai nepavyko, nei perkrovus projektą, nei perkrovus programą. Tik iš naujo sukompiliavus projektą buvo sėkmingai prisijungta su naujais vartotojais, patvirtintas aliarmas.

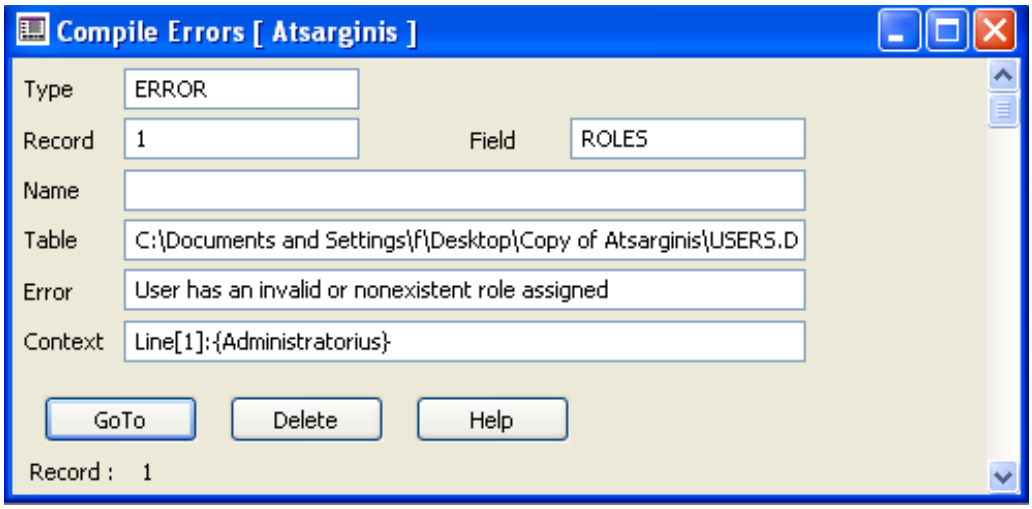
*Penktas variantas*. Įklijavus pakeistas bylas, kai programa išjungta prisijungti su naujais vartotojais ir slaptažodžiais nepavyko, galiojo seni. Visgi perkompiliavus projektą, pradeda galioti nauji vartotojai ir slaptažodžiai, patvirtintas aliarmas.

*Šeštas variantas*. Įklijavus naujas bylas, prisijungti naujais vartotojais nepavyko. Perkrovus projektą, senais vartotojais prisijungti nepavyko, tačiau įvedus naujo vartotojo duomenis, programos darbas sutriko, ir išmetė pranešimą, pateiktą 4.12 pav.




**4.12 pav.** Sisteminė klaida, pakeitus vartotojų bylų duomenis iš atskiro projekto.

Sukompiliavus projektą, išmetė klaidą – vartotojas priskirta negalima ar neegzistuojanti rolė, kuri pavaizduota 4.13 pav.




**4.13 pav.** Sisteminė klaida, kompiliuojant projektą po sisteminės klaidos.

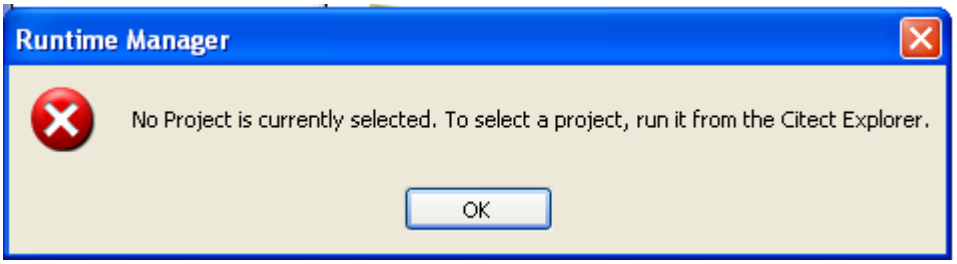
Priskyrus naujiems vartotojams roles, sėkmingai prisijungta projekte, patvirtintas aliarmas. Perkrovus programą – taip pat. Toliau bandoma sukurti kitame kompiuteryje naujus vartotojus, su tokio pat pavadinimo rolėmis, priskirtomis esamiems vartotojams, tik su visomis privilegijomis. Rezultatas – sėkmingai sukompiliuotas projektas, prisijungta su naujais vartotojais ir sėkmingai patvirtintas aliarmas.

*Septintas variantas*. Įklijavus naujas bylas, projektą sukompiliavus, sėkmingai prisijungta su naujais vartotojais ir patvirtintas aliarmas.

Tiriant bylų ištrinimo įtaką, iš pradžių bandoma ištrinti bylas, susijusias su projektu, runtime aplinkoje. Rezultatas – ištrintas visas projektas, nepavyko ištrinti tik aplankalo, susijusio su projektu. Sistema išmetė klaidą, pateiktą 4.14 pav.

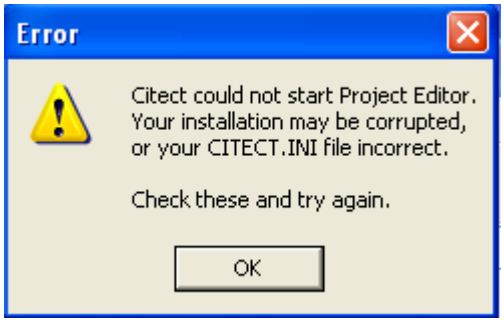

**4.14 pav.** Sisteminė klaida, bandant ištrintį visą aplankalą susijusį su projektu.

Projektas runtime aplinkoje toliau veikė, jokių sisteminių klaidų neišmetė, tačiau į operatoriaus komandas sistema neberegavo. Išjungus projektą runtime aplinkoje, projektas iš projektų sąrašo nedingo, tačiau bandant jį vėl paleisti, sistema išmetė klaidą, pateiktą 4.15 pav. Sistemos darbas buvo **sėkmingai** sutrikdytas.




**4.15 pav.** Sisteminė klaida, bandant paleisti ištrintą projektą.

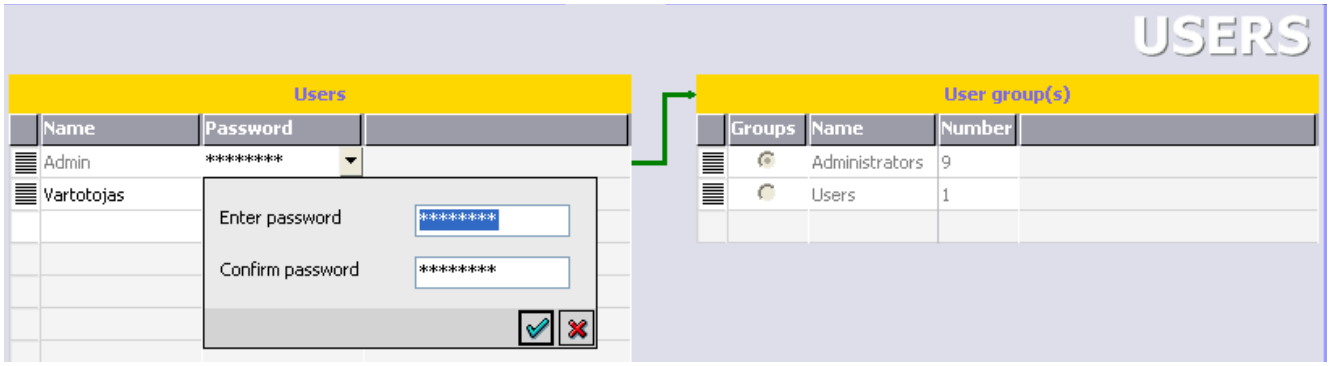
Sekantis žingsnis išsaiškinti galimybę ištrinti su programa susijusias bylas. Paleidžiamas projektas runtime aplinkoje, ir trinamos bylos, susijusios su CitectSCADA sistema. Ištrynus 823 iš 1027 bylas, išjungus projektą runtime aplinkoje, jo paleisti ši naujo nepavyko. Sistema nemetė jokių klaidų, tačiau bandant apleisti projektą nieko neįvykdavo – programa nereaguodavo. Paleidžiant programą iš naujo, programa nebepasileido – sistema išmetė klaidą, pateiktą 4.16 pav. Rezultatas – sistemos darbas **sėkmingai** sutrikdytas.




**4.16 pav**. Sisteminė klaida, ištrynus su programa susijusias bylas

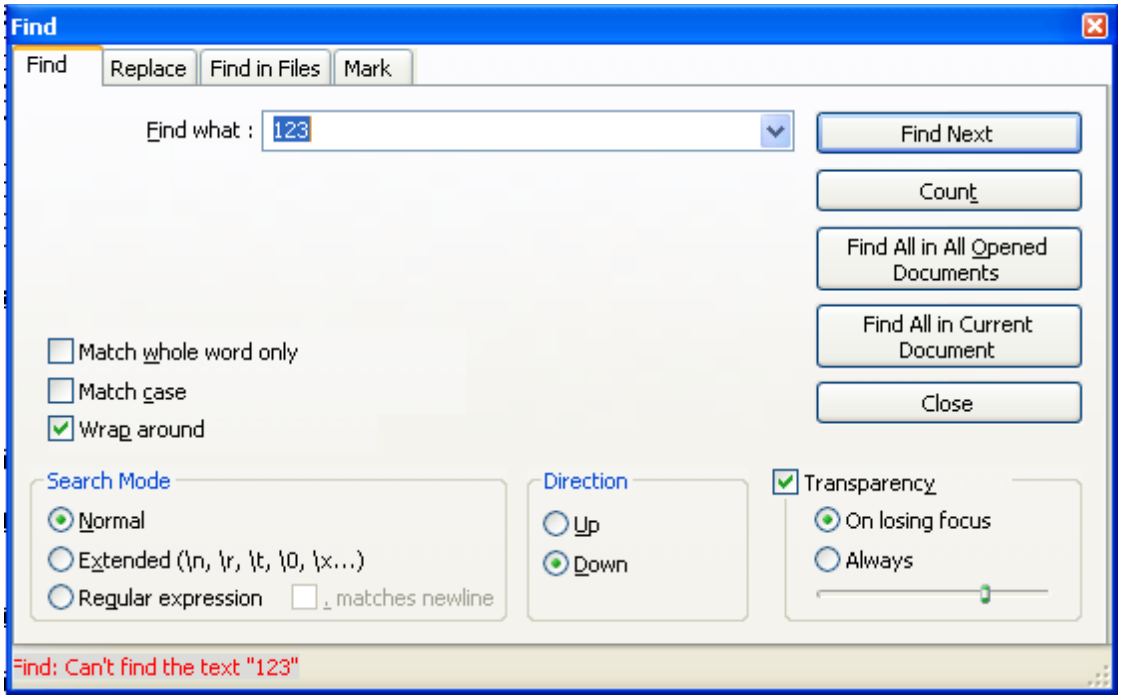
### 4.1.3. WinCC sistema

Sukuriamas projektas su vartotojų prisijungimui skirtu objektu (enhanced objects), su dviem vartotojų grupėmis – administrators ir users. Administrators grupei priskiriamas Adminas su slaptažodžiu 123, users grupei – vartotojas su slaptažodžiu 456. Programos vartotojų konfigūravimo langas su sukurtais vartotojais pateiktas 4.14 pav.




4.14 pav. WinCC vartotojų kūrimo langas.

Neradus informacijos, kur yra saugomi projekte sukurtų vartotojų duomenys, notepad++ programa atveriami visos projekte sukurtos bylos. Ieškoma simbolių kombinacijų : 123; 456; adm; use; var. Rezultatas pateiktas 4.15 pav.




**4.15 pav**. Vartotojo duomenų paieškos rezultatai WinCC projekto bylose.

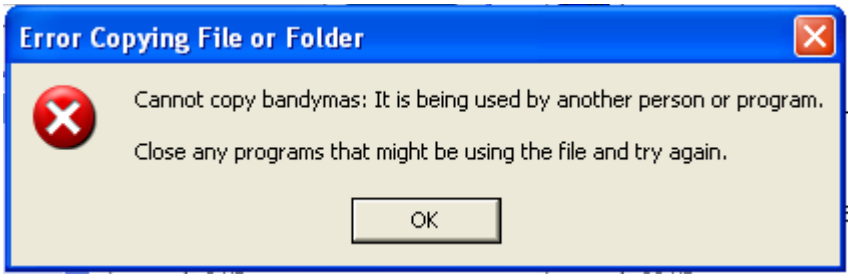
Kaip matoma, visi duomenys yra užšifruoti, ir jokios frazės, susijusios su ieškomais kriterijais rasti nepavyko. Atsižvelgiant į tai, kad atvėrus projekto vartotojų konfigūravimo langą (4.14 pav.), nukopijavus projektą į kitą kompiuterį, nei programos, nei bylų atvėrimo pagalba vartotojų duomenų nepavyko atskleisti, daroma išvada, kad duomenys **sėkmingai** užšifruoti.

Toliau tiriama galimybė projekto nukopijavimo, vartotojo duomenų pakeitimo kitame kompiuteryje ir jo įklijavimo atgal galimybė. Pakeičiamas administratoriaus slaptažodis į 111, vartotojo į 222, ir sukuriamas naujas vartotojas kazkas su slaptazodziu nepavyko ir priskiriamas naujai vartotojų grupei – power users.

Toliau tiriant vartotojų autentifikavimo apsaugą, tiriami šie variantai:

1. Įklijuojamos visos, kitame kompiuteryje pakeistos, projekto bylos runtime aplinkoje;
2. Įklijuojamos visos, kitame kompiuteryje pakeistos, projekto bylos development aplinkoje;
3. Įklijuojami visos, kitame kompiuteryje pakeistos, projekto bylos programai esant išjungtai.

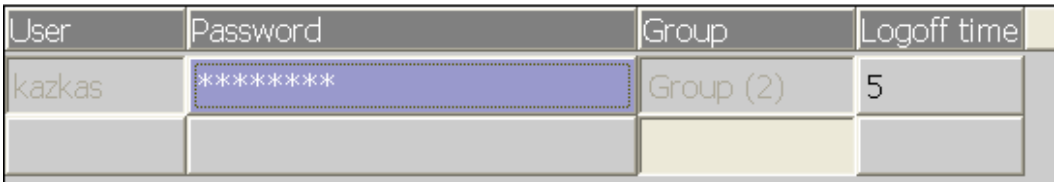
*Pirmasis variantas*. Pakeisti projekto bylų nepavyko, nes išmetė sisteminę klaidą, pateiktą 4.16 pav.




**4.16 pav.** Sisteminė klaida bandant pakeisti projekto bylas.

*Antrasis variantas*. Bandant pakeisti projekto bylas išjungus runtime aplinką, sistema išmetė tokią pat klaidą, kaip pateikta 4.16 pav.

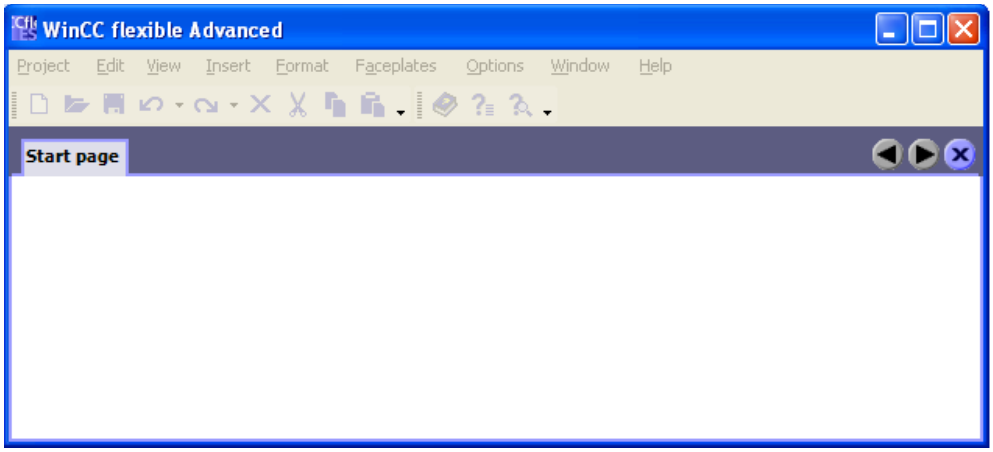
*Trečiasis variantas*. Įklijavus naujas bylas, projekte **sėkmingai** prisijungta su nauju vartotoju, bei esamais vartotojais, tačiau naujais slaptažodžiais. Sėkmingas prisijungimas pavaizduotas 4.17 pav.




**4.17 pav.** Sėkmingas prisijungimas WinCC aplinkoje.

Toliau tiriama su projektu susijusiu bylų trynimo įtaka. Visų su projektu susijusių bylų nepavyko ištrinti, tačiau pavyko ištrinti 9/12. Darbas runtime aplinkoje nesutriko, tačiau iš naujo paleisti projekto nebepavyko – programa nereagavo į paleidimą, tačiau klaidų taip pat neišmetė. Perkrovus programą, projektas sėkmingai pasileido, ir darbas toliau sėkmingai vyko. Šiuo atveju sistemos darbo sutrikdymas buvo **nesėkmingas**.

Trinant su programa susijusias bylas runtime aplinkoje, pavyko ištrinti 23 iš 28 tūkstančių bylų. Rezultatas – perkrauti projekto nebepavyko, sistemos darbas sėkmingai **sutrikdytas**. Perkrauti sistemą pavyko tik dalinai – pasileido programa, tačiau joje visos funkcijos buvo panaikintos. Lango pavyzdys pateiktas 4.18 pav.




**4.18 pav**. Sutrikdytos WinCC sistemos programos langas.

## 4.2. OS lygmuo

Tiriant vartotojų saugumą taikant OS lygmens apsaugas, yra naudojamos trijų lygių Windows OS vartotojų grupės – administrators, users, power users.

### 4.2.1. InTouch sistema

Sukuriamas programinis kodas (scriptas), vartotojų grupėmis nuskaityti ir suteikti jiems skirtingus prieigos lygius (Access levels), kuris suveikia tik paleidus projektą runtime aplinkoje (on startup):

*DIM Assigned AS DISCRETE;*

*Assigned = AddPermission("bandymas", "administrators",1000);*
*Assigned = AddPermission("bandymas", "power users",9999);*
*Assigned = AddPermission("bandymas", "users",5000);*

Toliau sukuriamas Memory message tipo kintamasis GroupMembership, ir parašomas Data change programinis kodas (scriptas):

```
DIM Member AS DISCRETE;
GroupMembership = "No Group Membership";

Member = QueryGroupMembership( "bandymas", "administrators");
IF Member == 1 THEN
GroupMembership = "Kompiuterio administratorius";
ENDIF;

Member = QueryGroupMembership( "bandymas", "users");
IF Member == 1 THEN
GroupMembership = "Vartotojas";
ENDIF;

Member = QueryGroupMembership( "bandymas", "power users");
IF Member == 1 THEN
GroupMembership = "Meistras";
ENDIF;
```

Šio skripto pagalba įgalinamas atvaizdavimas suteiktos prisijungimo grupės lygmuo. Toliau tiriami šie atvejai:

1. Prisijungimas su naujai sukurtu vartotoju;
2. Vartotojo, priklausančio kelioms grupėms, prieigos lygio suteikimas;
3. Esamų vartotojų grupės keitimas.

*Pirmasis atvejis*. Projektas veikia runtime rėžimu. Sukuriamas naujas vartotojas As priklausantis administrators grupei. Prisijungimas – **sėkmingas**.

*Antrasis atvejis*. Sukuriamas naujas vartotojas As, priklausantis administrators ir users vartotojų grupėm. Prisijungus suteiktas prieigos lygmuo – vartotojas. Toliau nagrinėjamas scriptų veikimas. Pirmame scripte sukeičiama vietomis administratoriaus ir vartotojų grupių prieigos lygių priskyrimas, o antrame nieko nekeičiama, ir pakartojamas prisijungimas. Rezultatas – suteiktas priėjimo lygmuo **vartotojas**. Atstatoma pirmojo scripto struktūra, ir sukeičiama vietomis antrojo scripto Groupmembership kintamojo eilutės. Rezultatas - suteiktas priėjimo lygmuo **aplikacijos administratorius**.

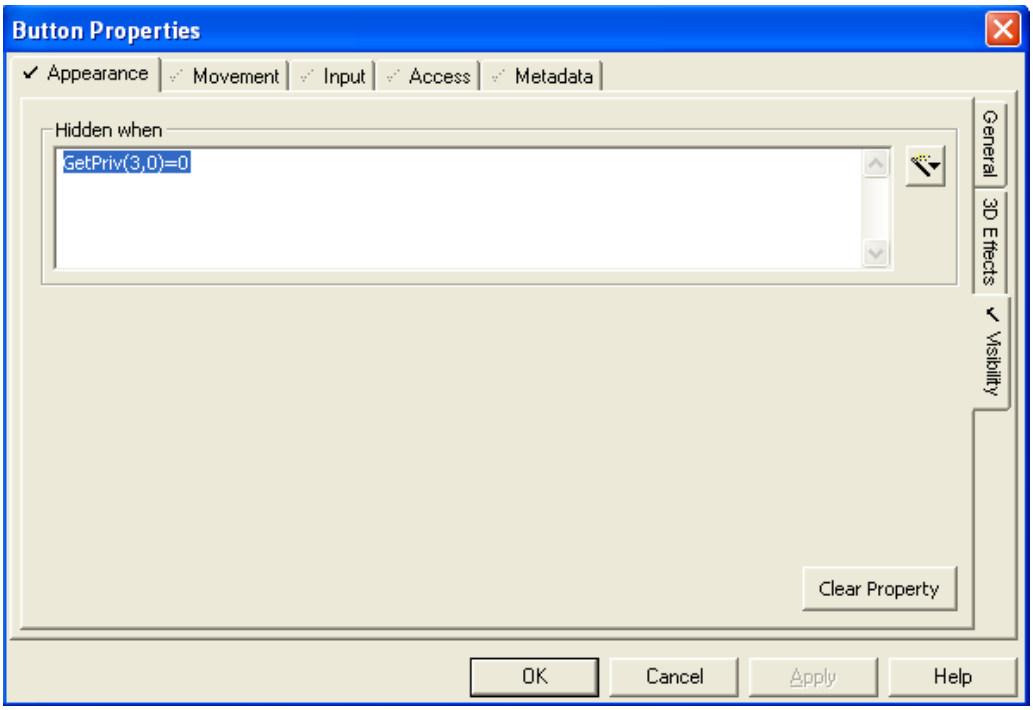
*Trečiasis atvejis*. Sukariamas vartotojas Tu su users vartotojų grupe. Pakeičiama jo grupė į power users, rezultatas – suteiktas **meistras** priėjimo lygmuo. Pastaba: jei atliekamas pakeitimas, tačiau su vartotoju neprisijungiama iš naujo, pakeitimai įtakos **neturės**.

### 4.2.2. CitectSCADA sistema

Atsižvelgiant į tai, kad taikant OS lygmens apsaugą, programa priskiria vartotojus grupėms, su atitinkamomis privilegijomis, sukuriami trys vartotojų tipai:

a) Vartotojas, priklausantis users vartotojų grupei, ir turintis 2 lygio privilegijas;
b) Administratorius, priklausantis administrators vartotojų grupei, ir turintis 1 lygio privilegijas;
c) Operatorius, priklausantis power users vartotojų grupei, ir turintis 3 lygio privilegijas.

Tam, kad išsiaiškinti koks yra suteiktas privilegijų lygis prisijungus, projekte sukuriami trys klavišai, kurių matomumas priklauso nuo privilegijų lygmens. Power users tipo vartotojam sukurto klavišo pavyzdys pateiktas 23 pav.




**4.19 pav.** Power users privilegijų grupei priskirtas klavišas.

Sukūrus klavišus ir vartotojus, bandant prisijungti tiek su administratoriumi, tiek su operatoriumi, taip pat priskiriama ir user vartotojų grupės privilegija. Daroma prielaida, kad kuriant naujus vartotojus, jie iš pradžių priskiriami users vartotojų grupei, ir tik vėliau priskiriami kitai, o iš users vartotojų grupės pašalinus, CitectSCADA sistemoje išlieka vis tiek taip pat, kaip ir paprastas vartotojas.

Perkrovus sistemą, perkompiliavus projektą – **pokyčio nėra**. Priskyrus users grupės privilegijas backup operators grupei, ir perkėlus vartotoją į naują grupę, privilegijos nebepriskiriamos users grupei.

Toliau tiriami šie atvejai:

1. Prisijungimas su naujai sukurtu vartotoju;
2. Vartotojo, priklausančio kelioms grupėms, prieigos lygio suteikimas;
3. Esamų vartotojų grupės keitimas.

*Pirmasis atvejis.* Projektui veikiant runtime aplinkoje, sukuriamas naujas vartotojas As, kuris automatiškai priskiriamas users vartotojų grupei. Prisijungimas – **sėkmingas**.

*Antrasis atvejis*. Sukuriamas vartotojas mes, kuris priskiriamas administrators ir power users vartotojų grupėms. Rezultatas –suteikti administrators ir power users vartotojų priėjimo lygmenys.

*Trečiasis atvejis.* Sukuriamas vartotojas Tu, ir pakeičiama jo grupė į power users. Rezultatas – **suteikta**s operatoriaus priėjimo lygmuo, kartu su users lygmeniu.

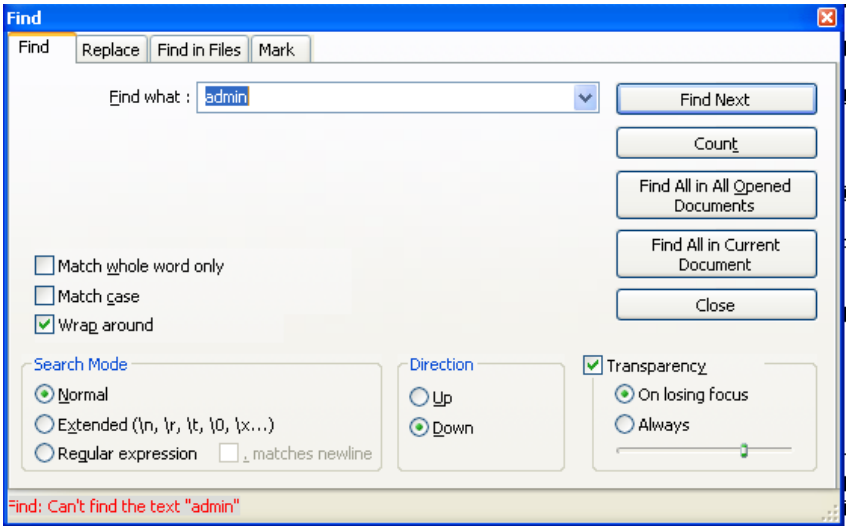
## 4.3. Nuotolinės duomenų bazės lygmuo

### 4.3.1. ArchestrA IDE

Nuotolinės duomenų bazės saugumo tyrimui pasirenku ArchestrA IDE programą, kurioje sukuriama merged tipo aplikacija. Šioje aplikacijoje vartotojų duomenų bazės konfigūravimas vyksta ArchestrA IDE aplinkoje, tačiau pačio projekto vizualicijos kūrimas vyksta InTouch WindowMaker aplinkoje. Toliau sukuriami du vartotojai:

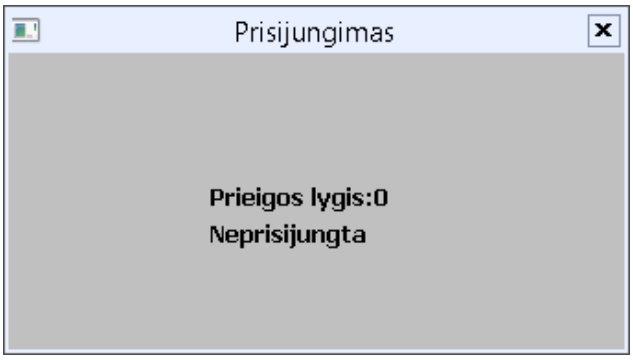
1) vartotojas admin su slaptažodžiu admin, priklausantis administratorius vartotojų grupei, turinčiai prieigos lygį 9999;
2) vartotojas operatorius, priklausantis operatorius vartotojų grupei, turinčiai prieigos lygį 6500.

Toliau bandoma dešifruoti sukurtų vartotojų duomenis. Atsižvelgiant į informacijos apie bylą, kurioje saugomi vartotojų duomenys, nebuvimą, visas sukurtas projektas (galaxy) yra išsaugomas vienoje byloje su plėtiniu .cab, pasitelkiant backup funkciją sistemos valdymo konsolėje (system management console). Byla atveriama tiek su notepad++, tiek su hex editor ir ieškoma frazės „admin“. Rezultatas – frazė nerasta, pateikta 4.20 pav.



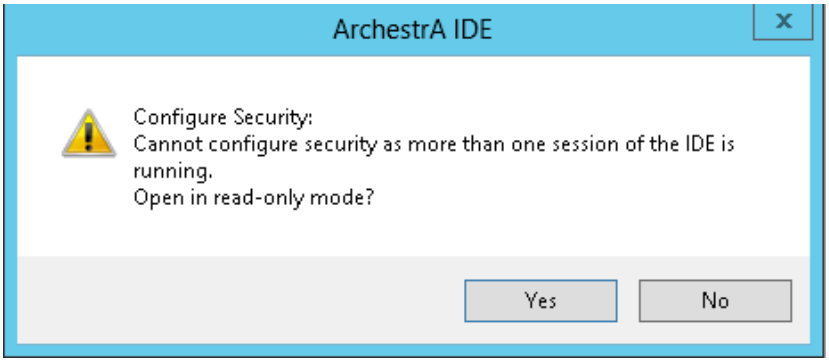

**4.20 pav.** Frazės „admin“ paieška notepad++ programoje.

Aplikacijoje sukuriamas langas su teksto lauku „prieigos lygis:“ ir prieigos lygio atvaizdavimo funkcija $Accesslevel bei tekstu, informuojančiu, jog neprisijungta, jei prieigos lygis yra 0. Lango pavyzdys pateiktas 4.21 pav.




**4.21 pav.** Prisijungimo lygio atvaizdavimo langas WindowViewer aplinkoje.

Toliau bandoma pakeisti esamus vartotojus ArchestrA IDE aplinkoje, aplikacijai veikiant WindowViewer aplinkoje. To padaryti nepavyko, sistema išmetė klaidą, pateiktą 4.22 pav.

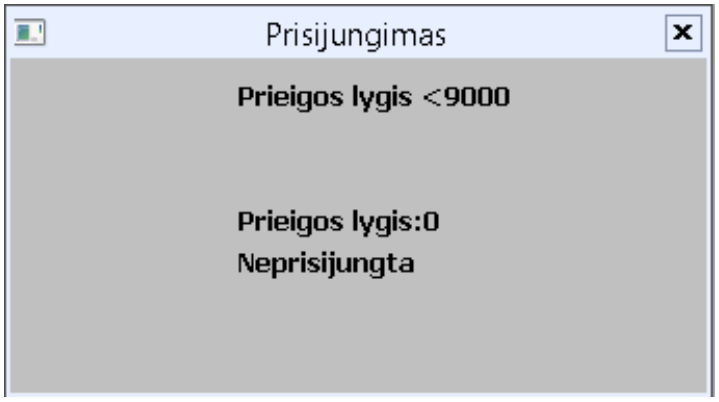



**4.22 pav**. Programinė klaida, bandant pakeisti vartotojų duomenis veikiant aplikacijai.

Norint pakeisti, būtina išjungti aplikaciją tiek WindowViewer, tiek WindowMaker aplinkose. Atsižvelgiant į šį faktą, belieka ištirti, kokį prieigos lygį suteikia sistema. Tam sukuriami papildomi tekstiniai:

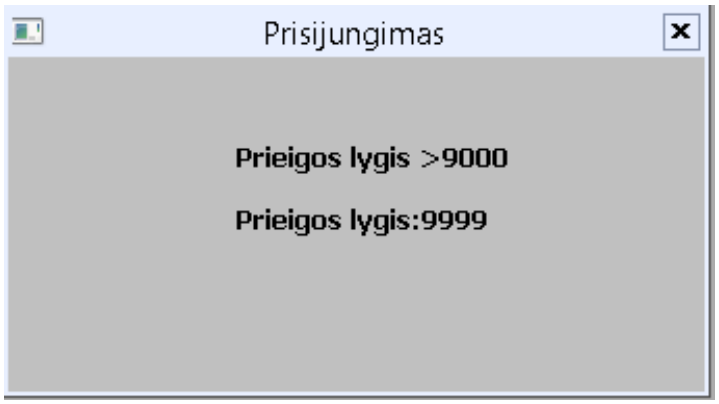
1) Prieigos lygis <9000, kai vartotojui suteiktas prieigos lygis mažiau 9000;
2) Prieigos lygis >9000, kai vartotojui suteiktas prieigos lygis daugiau 9000.

Papildytas aplikacijos langas pateiktas 4.23 pav.




**4.23 pav**. Papildytas prisijungimo lygio atvaizdavimo langas WindowViewer aplinkoje.

Toliau pakeičiama vartotojo operatorius grupių priklausomybė – jis taip pat priskiriamas ir administrators vartotojų grupei. Bandoma prisijungi, rezultatas (4.24 pav) – priskirtas aukščiausias prisijungimo lygis, priklausantis administrators vartotojų grupei.




**4.24 pav**. Suteiktas prieigos lygis prisijungus su vartotoju, priklausančiu keliom vartotojų grupėm

# Rezultatai

1. Visi šiame darbe nagrinėti supervizorinio valdymo ir duomenų surinkimo taikomųjų programų apsaugos posistemiai sėkmingai užšifruoja vartotojo slaptažodžius. Atvėrus su jais susijusias bylas tiek hex editor, tiek notepad programa, duomenys susiję su vartotojais pilnai užšifruoti tiek InTouch, tiek WinCC sistemose, tuo tarpu CitectSCADA sistemoje pavyko dešifruoti vartotojų vardus;
2. InTouch sistemoje, aplikacijos lygmenyje įklijuojant pakeistą password.bin bylą iš visiškai kito projekto, prisijungimas yra sėkmingas su naujais vartotojais;
3. CitectSCADA sistemoje įklijavus pakeistas _Users.rdb ir Users.dbf bylas prisijungta sėkmingai tik perkompiliavus projektą;
4. WinCC programoje reikia pakeisti visą projektą, norint atlikti pakeitimus vartotojams, ir tai įmanoma atlikti tik išjungus programą;
5. Sukūrus naują vartotoją, InTouch ir CitectSCADA sistemose prisijungimas galimas iš karto;
6. Sukūrus vartotoją, priklausantį keliom operacinės sistemos vartotojų grupėm, InTouch sistemoje priskiriama privilegija priklauso nuo programinio kodo (scripto) struktūros, CitectSCADA sistemoje priskiriamos visos tam vartotojui priskirtu grupių privilegijos.
7. Pakeitus vartotojo grupę, InTouch sistemoje rezultatas įsigali tik iš naujo prisijungus su vartotoju, CitectSCADA sistemoje taip pat.
8. CitectSCADA sistemoje sukūrus vartotoją ne users grupėje, jam vistiek priskiriamos users grupės privilegijos;
9. InTouch ir CitectSCADA sistemose ištrynus bylas, susijusias su projektu, jam veikiant runtime aplinkoje, pakartotinis projekto paleidimas negalimas. Tik WinCC sistemoje projektą galima paleisti iš naujo;
10. Visose sistemose ištrinant bylas, susijusias su vykdomąja programa, sistemos darbas sėkmingai sutrikdomas;
11. ArchestrA IDE sistemoje vartotojų duomenys sėkmingai užšifruoti;
12. ArchestrA IDE sistemoje, vartotojam priskiriamos rivilegijos priklauso nuo jam priklausančių grupių didžiausio priėjimo lygmens.

# Išvados

1. Saugiausios sistemos vartotojų duomenų užšifravimo atžvilgiu – InTouch ir WinCC, jose nepavyko nuskaityti jokių duomenų. CitectSCADA bylose pavyko nuskaityti esamų vartotojų vardus;
2. Saugiausia sistema keičiant su vartotojais susijusias bylas – WinCC, nes reikia perkrauti visą programą iš naujo, tuo tarpu CitectSCADA aplinkoje – projektą pakanka tik perkrauti, programos paleisti iš naujo nepareikia. InTouch sistemoje, įklijavus pakeistas bylas, projekto perkrauti net nereikėjo;
3. Operacinės sistemos (OS) apsaugos lygiu visos programos yra vienodai saugios – visose prisijungimas galimas iškarto atlikus pakeitimus OS vartotojų duomenų bazėje;
4. ArchestrA IDE vartotojų duomenų bazės konfigūravimas problematiškas – būtina atsijungti nuo aplikacijos, norint atlikti pakeitimus vartotojų duomenų bazėje;
5. Saugiausia sistema su projektu susijusių bylų pašalinimu atžvilgiu – WinCC. Kol sistema veikia, bylų ištrinti nepavyksta, o ištrynus visas, išskyrus pagrindinę, jos sėkmingai atstatomos iš naujo perkompiliavus projektą. Tuo tarpu InTouch ir CitectSCADA sistemose su projektu susijusių bylų ištrynimas sėkmingai sutrikdo sistemos veikimą – sekantį kartą projekto paleisti nepavyksta;
6. Visos sistemos vienodai pažeidžiamos, kai trinamos su programa susijusios bylos – projektas toliau veikia, tačiau nei programos, nei projekto pakartotinai paleisti nebepavyksta.

# Literatūros šaltiniai


1. InTouch User Manual;

2. CitectSCADA User Manual;

3. ArchestrA Integrated Development Environment (IDE) User‘s Guide. Žiūrėta 2016-03-23. Prieiga per internetą: http://platforma.astor.com.pl/files/getfile/id/1286;

4. WinCC flexible 2008 User‘s Manual. Žiūrėta 2016-03-23. Prieiga per internetą: https://cache.industry.siemens.com/dl/files/010/18796010/att_99844/v1/Users_Manual_WinCC_flexible_en-US.pdf ;

5. Internal security attacks on SCADA systems, 2013. Žiūrėta 2017-01-02. Prieiga per internetą: http://ieeexplore.ieee.org/xpls/icp.jsp?arnumber=6579516 ;

6. IP fragmentation attack. Žiūrėta 2017-01-02. Prieiga per internetą: https://www.incapsula.com/ddos/attack-glossary/ip-fragmentation-attack-teardrop.html ;

7. Netwox tool. Žiūrėta 2017-01-02. Prieiga per internetą: http://ntwox.sourceforge.net/ ;

8. LOIC. Žiūrėta 2017-01-02. Prieiga per internetą: https://sourceforge.net/projects/loic/ ;

9. SCADA System Cyber Security – A comparison of Standards. Žiūrėta 2017-01-02. Prieiga per internetą: https://scadahacker.com/library/Documents/Standards/IEEE%20-%20Comparison%20of%20SCADA%20Security%20Standards.pdf ;

10. Guide to Industrial Control Systems (ICS) Security. US National institute of standards and technology. Žiūrėta 2017-01-02. Prieiga per internetą: http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-82r2.pdf ;

11. SCADA security in the light of Cyber-Warfare. Žiūrėta 2017-01-02. Prieiga per internetą: http://www.sciencedirect.com/science/article/pii/S0167404812000429 ;

12. FactorySuite and ArchestrA InTouch® Application File List. Žiūrėta 2017-01-15. Prieiga per internetą: https://wonderwarewest.com/download/Wonderware%20Tech%20Notes/0145%20FactorySuite%20and%20ArchestrA%20InTouch%C2%AE%20Application%20File%20List.pdf ;

13. Hex editor neo. Prieiga per internetą: http://www.hhdsoftware.com/hex-editor ;

14. Insource solutions. How to reset the administrator password intouch. Žiūrėta 2017-01-25. Prieiga per internetą: https://insource.mindtouch.us/Support_Tickets/InTouch/Installation%2F%2FConfiguration/060765_-_How_to_reset_the_administrator_password_in_InTouch