Balsavimas internetu – saugu, o keliamos rizikos – nepagrįstos

Moksliniu tyrimu paremta analizė

Analizę parengė kompiuterių mokslo ir IT saugumo ekspertas:

Kęstutis Matuliauskas

2015 sausio 15

Turinys

ĮVADAS

1 pav. Balsavimas internetu - jo rizikos ir privalumai

Balsavimas internetu yra saugus tinkamai paruoštoje sistemoje, o jo rizikos yra ne didesnės nei paprasto popierinio balsavimo. Išanalizavau balsavimo internetu kritikų pateikiamus nuogąstavimus, ir detaliai atsakau, kodėl šios **rizikos yra nepagrįstos ir suvaldomos**. Dauguma grėsmių yra cituojamos iš 10-15 metų senumo straipsnių, t.y. laikų, kai žmonės naudojo "Internet Explorer 6″, "ActiveX" įskiepiais, "Windows XP" operacine sistema, kurioje nebuvo integruotų ugniasienių, antivirusinių programų ir vartotojo teisių valdymo (angl. "UAC – user account control"). O šališkų ekspertų vaizdo klipuose yra iškeliamos tos rizikos, kurios praktikoje sudaro mažiau nei pusę procento balsų, t.y. dabar esanti popierinių balsų papirkinėjimo rizika yra žymiai didesnė, nei būtų balsuojant internetu.

Kritikų cituojami straipsniai yra 11 metų senumo, daryti 2004 m. JAV gynybos departamento užsakymu. Negana to – JAV tikrai nėra ta šalis, kuri yra tinkamas pavyzdys el. inovacijoms – ten žmonės dar gausiai naudojasi čekiais apmokėjimams gauti ir ten vis dar aktuali čekių klastojimo problema, o paprasti ir lengvi internetiniai pervedimai, tarp kelių banku planuojami tik po 10 metų, kai Lietuvoje ir kitose Europos šalyse jie egzistuoja jau daugiau nei 10 metų. O IT eksperto Bruce Schneier straipsnis yra dar senesnis – 2000-ųjų metų – jam net 15 metų. Negana to, pačių internetinio balsavimo kritikų Lietuvoje darytos analizės yra taip pat beveik 10 metų senumo, ir rašytos dar 2006-ais metais. Per tą laiko tarpą interneto kokybės standartai pasikeitė radikaliai, o kietųjų diskų talpa išaugo net 1000 kartų. Taip pat per tą laiką buvo patentuotos naujos el. saugumo technologijos, kaip "kompiuterinis klavišų parašas" (angl. "Keystroke Dynamics"), atlikti tokių sistemų bandymai praktikoje, atrasti ir pašalinti didžiausi šių sistemų jų trūkumai, ir saugi balsavimo internetu sistema tapo realybe.

Ši analizė parašyta remiantis Estijos balsavimo internetu sistema, įtraukiant gerąsias praktikas iš Norvegijoje naudotos balsavimo internetu sistemos, bei pateikiant siūlomus pakeitimus ir sprendimus panaikinti rizikoms tokią sistemą kuriant ar perkant ir modifikuojant, Lietuvoje.

1. Rizika – įsilaužimas į sistemą (balso duomenų paviešinimas)

1.1. Rizika ir jos sprendimas

Baimė - *‘sistemą nulauš “hakeriai” ir išviešins kas už ką balsavo’*

Išvada – saugioje sistemoje ši rizika mažesnė nei 0,001%. O ‘pavogta’ duomenų bazė būtų nieko verta ir neiššifruojama.

1.2. Kaip užtikrinti sistemos duomenų saugumą?

Atsakymas – sistema turi būti padaryta tai, kad net ir turint jos kopiją nebūtų galima iššifruoti žmogaus duomenų. Kiekviena šių dienų sistema turi būti kuriama naudojant vienkrypčius šifravimo algoritmus – tarkim, kad ir **sha256** algoritmą, persukus tarkim **9,563 kartus** (interacijas), tai yra – darant vadinamąjį “**key-stretching**“. Arba naudojant **scrypt**, **bcrypt**, ar **PBKDF2** algoritmą (angl. „Password-Based Key Derivation Function 2“), besiremiantį RSA kriptografijos standartu. Šių algoritmų pagalba užtikriname, kad įsilaužėliai negalės pasinaudoti milžiniškomis slaptažodžių lentelės (angl. “ Rainbow Table”), kurios paprastai būna 200 GiB – 20 TiB dydžio ar didesnės. Šios lentelės – tai slaptažodžių ir sugeneruotos jų maišos rezultatų “hash” duomenų bazė, kurioje paprastai saugomi 1 ar 2 slaptažodžio iteracijos. “Key-stretching” dėka – tarkim 9,563 iteracijų – mes ne tik žymiai sulėtiname slaptažodžio generavimo laiką, bet kartu itin apsunkiname neįmanoma slaptažodžių lentelės (angl. “Rainbow Table”) panaudojimą.

Pasitelkus dalį (angl. „salt“) el. parašo rakto, duomenys galėtų būti šifruojami žemiau nurodyti tokiu principu ir saugomi duomenų bazėje. Pseudo-kodas:

```
// RSA key is in the ID Card with E-Signature
$RSAKey = "%^_S$G$AFA$S445566asrgo$552-+";
$salt = getRSASalt($RSAKey);
$valueToSave = scrypt("Jonas.Pavardaitis.38905200189".$salt);
$ORM->startTransaction();
$ORM->transactionSetVoted("Jonas", "Pavardaitis", "38905200189");
$ORM->transactionInsertVote($valueToSave, "Person 1", time());
$ORM->commit();
```

Na o patikrinimas, ar balsas nebuvo pakeistas, galėtų būti realizuotas duomenų bazėje įvykdant paiešką šifruotas reikšmes - duomenų bazėje. Pseudo-kodas:

```
// We are in ‚Try‘ block. SQL to ORM transformation shouldn‘t be applied
$currentVote = "Person 1";
$ormTransformed = makeORMQuery("SELECT vote FROM votes_table WHERE vote_data=’
".$newGeneratedData."’");
$oldVote = $instanceORM->getData($ormTransformed);

// IF "Person 1" not equals to "Person 1"
if($currentVote != $oldVote)
{
   throw( "Tavo balsas pakeistas" );
}
```

1.3. Kiek laiko užtruktų iškoduoti gerai užšifruotus duomenis?

Atsakymas – su greičiausiais pasaulio super-kompiuteriais vos vienam balsui iššifruoti prireiktų 10,000 metų, o su kvantiniais kompiuteriais – apie 200 metų. Tai bet kuriuo atveju viršija žmogaus gyvenimo amžių. O pats rinkimų procesas trunka tik savaitę laiką.

1.4. Kokia yra šios rizikos (duomenų atskleidimo) tikimybė?

Atsakymas – 0,001% arba rizika nepagrįsta. Visų pirma – reikėtų sugebėti įsilaužti į tokią saugią sistemą, o tokio įsilaužimo rizika yra 0,1%. O antra – įsilaužus tuos duomenis dešifruoti – tad bendra rizika yra mažesnė už 0,001%. Negana to – yra siūloma praėjus savaitei ar pusmečiui po rinkimų apskritai pašalinti algoritmais užšifruotą rinkėjo informaciją (vardą, pavardę, asmens kodą) iš duomenų bazės įvykdant, tarkim "***DROP COLUMN*** voters_data" komandą. Tuomet duomenų bazėje išlinks tik Eilės NR., balsas (už ką balsuota), ir balso data UNIX_TIMESTAMP() formatu, t.y. sisteminiu laiku sekundėmis nuo 1970 m. sausio 1 d., nuo kada yra skaičiuojamas UNIX laikas visose pasaulio kompiuterinėse sistemose, kuris yra nuolat sutikrinamas su pasaulinėmis laiko tarnybomis, besinaudojančiomis tiksliausiais pasaulyje – atominiais laikrodžiais (angl. "atomic watch").

2. Rizika – neįmanoma užtikrinti duomenų anonimiškumo

2.1. Rizika ir jos sprendimas

Baimė - *"balsavimo komisija žinos kas už ką balsuoja ir perduos šią informaciją partijoms ir 'blogiečiams'"*

Išvada – duomenys privalo būtų užšifruoti vienkrypčių algoritmu, kad tokią pačią reikšmę sugeneruoti ir balsą pasitikrinti galėtų tik pats vartotojas žinodamas savo PIN kodą. Nuo "keyloggerių" gali būtų saugomasi programiškai uždraudžiant (programiškai patikrinus) balsuoti iš saugumo kriterijų neatitinkančių kompiuterių, pavyzdžiui – naudojančių "Windows XP" OS, "Internet Explorer 6″ interneto naršyklę.

2.2. Kaip užtikrinti duomenų anonimiškumą?

Atsakymas – sistemoje visi duomenys turi būti šifruojami vienkrypčiu algoritmu, kurio analogišką reikšmę tik tam tikroje vietoje ir tam tikru metu gali sugeneruoti tik pats balsuotojas, bet ne kitas žmogus. Pvz. žmogus, gali pasitikrinti savo balsą kitame įrenginyje – telefone – 30 sekundžių po balsavimo, pamatyti pagal QR kodą savo balsą. Tik žmogus, kuris balsavo, turi "raktą", tad tik jis gali savo balsą patikrinti, kad jis nebuvo pakeistas. Kiti žmonės kas už ką balsavo sužinoti negali, nes duomenys yra užšifruoti vienkrypčiu algoritmu.

Virusų poveikio rizika, esant bent dviem įrenginiams (kompiuteriui ir telefonui) yra labai maža – 0,1%, tad šią riziką naudingiausia pripažinti valdoma rizika ir ją prisiimti valstybei, kaip tai daroma Estijos atveju.

2.3. Kaip apsisaugoti nuo šnipinėjimo programų, įrašančių balsavimo istoriją (angl. "keyloggerių") ?

Atsakymas – galima reikalauti, kad balsuojančiojo asmens kompiuteris programiškai būtų atnaujintas iki tuo metu naujausios interneto naršyklės ir jo operacinėje sistemoje būtų įdiegti naujausi atnaujinimai, ar tai būtų „Windows Vista", „Windows 7", „Windows 8", „Windows 8.1″, „Ubuntu", „OS X" ar kita palaikoma operacine sistema (OS). Visos modernios OS turi integruotą nemokamą antivirusinę ir anti-šnipinėjimo programą (pvz. "Windows Defender"), o nuo įsilaužimu saugo nemokama "Windows Firewall" integruota ugniasienė, kurie gauna naujausią informaciją apie grėsmes kasdien ir atsinaujina kasdien. Antivirusinės programos yra sukurtos net ir Android OS, pvz. "MalwareBytes AntiMalware Mobile", tad galima užtikrinti net ir telefono įrenginio saugumą.

Patikrinimą, ar vartotojas turi naujausią OS, naršyklės versiją, antivirusinės programos atnaujinimus galima tiek serverio skriptų pagalbą, suteikiančių informaciją apie vartotojo kompiuterį, tiek ir Java aplikacijų pagalba. Labai bijant virusų rizikos, galima visiems vartotojams, kurių kompiuteriai neatitinka reikalavimų pateikti lentelė "Jūsų kompiuteris neatitinka programinės įrangos kokybės kriterijų", ir pastarieji žmonės turėtų tiesiog balsuoti eidami į balsavimo apylinkę ar ambasadoje, kaip yra ir iki šiol.

Tad nuo “keyloggerių” gali būtų saugomasi programiškai uždraudžiant (programiškai patikrinus) balsuoti iš saugumo kriterijų neatitinkančių kompiuterių, pavyzdžiui – naudojančių “Windows XP” OS, “Internet Explorer 6″ interneto naršyklę, ar kitą nesaugią programinę įrangą.

3. Rizika – balsų klastojimas serverio pusėje

3.1. Rizika ir jos sprendimas

Baimė – *"hakeriai" ar "papirktas programuotojas" pakeis balsus*

Išvada – sistema turi būti sukurta decentralizuotą, duomenys saugomi grupėje skirtingų serverių, skirtingose vietose ar debesyse (angl. "Cloud servers"), su skirtingais prie jų dirbančiais žmonėmis, ir serveriuose instaliuota skirtinga aparatūrinė (Intel ir AMD) ir programinė įranga ("Ubuntu Server", "Windows Server", "Solaris" ir kita)

3.2. Kaip apsisaugoti nuo balsų klastojimo e-balsavimo sistemoje?

Atsakymas – į šią problemą reikia žvelgti iš įsilaužimo perspektyvos. Tarkim buvo įsilaužta į serverį ir pakeisti balsai. Jeigu duomenys bus saugomi decentralizuotai, tai vadinasi kiti 9 serveriai praneš sistemos administratoriams, kad buvo įsilaužta į vieną iš serverių, ir pakeistas tam tikras balsas, todėl likusios sistemos tą pakeistą balsą neutralizuos ir paliks originalią jo reikšmę remiantis kitais 9 serveriais.

Patys serveriai turėtų būti išdėstyti skirtingose vietose:

- **Jei naudojami dedikuoti serveriai** – jie turėtų būti skirtingose fizinėse vietose ("Šiaulių duomenų centre", "Kauno duomenų centre" ir kitur). Serverius turi aptarnauti skirtingos įmonės, bei kiekvienoje iš serverių būstinių rinkimų metu, kiekviena iš partijų galėtų deleguoti po savo stebėtoją, visam rinkimų periodui, įskaitant ir budėjimą naktimis, taip užtikrinant sistemos prižiūrėtojų nešališkumą. Taip pat darbas kiekviename iš duomenų centrų turėtų būti filmuojamas visą laiką.
- **Jeigu naudojami virtualūs serveriai (VPS)** – jie turėtų būti skirtinguose Cloud sprendimuose ("Amazon AWS", Microsoft Azure", "Google Cloud" ir kitur), kuriuose virtualių serverių fizinė vieta būtų skirtingose pasaulio vietose (Airijoje, Didžiojoje Britanijoje, Vokietijoje, JAV ir kitur).

Naudojant virtualius serverius nebereikia patiems spręsti fizinės serverių Pastaruoju atveju jau šiuo metu egzistuoja itin aukšto lygio apsaugos fizinė infrastruktūra, apsauganti nuo fizinės prieigos prie serverių, leidimo neturintiems žmonėms, o tokie yra visi pašaliniai asmenys.

Svarbu, kad skirtinguose serveriuose būtų instaliuota skirtinga aparatūrinė ir programinė įranga:

- **Skirtinga aparatūrinė įranga** – Intel ir AMD procesoriais, skirtingos motininės plokštės su skirtingomis motininių plokščių tvarkyklėmis (angl. "Firmware")
- **Skirtinga programinė įranga** – "Ubuntu Server" OS, "Windows Server" OS, "Solaris" OS ir kita.

4. Rizika – balso suklastojimas kliento pusėje

4.1. Rizika ir jos sprendimas

Baimė – *"išanalizavę Estijos modelį, Mičigano universiteto (JAV) mokslininkai teigia aptikę teoriškai įmanomą galimybę, kad net ir patikrinus balsą telefone, virusu užkrėstas kompiuteris, kuriame virusas bus aktyvavęs "keyloggerį", ir jo pagalba išsaugos kliento PIN kodą, kliento kompiuteriu prabalsuos dar kartą už kitą kandidatą"*

Išvada – Mičigano mokslininkai nesusipažinę su trečiuoju saugumo lygiu, taikomu modernių Europos bankų el. bankininkystėje – elektroniniais kodų generatoriais (atskirais fiziniais įrenginiais). Kodų generatorius neutralizuoja riziką, kad balsą pakeis virusas.

4.2. Kaip apsisaugoti, kad virusas vartotojo kompiuteryje neprabalsuotų dar kartą?

Atsakymas – panašu, kad Mičigano mokslininkai nelabai susipažinę, kaip nuo "keyloggerių" ir fizinių PIN kodo ir slaptažodžio vagysčių saugosi modernūs bankai. Tas yra tiesa, nes JAV bankuose tokios apsaugos dar matyti neteko – ten norint prisijungti prie el. bankininkystės užtenka vartotojo vardo ir slaptažodžio. Tačiau Europoje ir ypač Lietuvoje jau seniai naudojama 3-iojo lygio apsauga norint prisijungti prie interneto bankininkystės – tai kodų kortelės, o šiuo metu jas jau baigia pakeisti dar saugesni instrumentai – elektroniniai kodų generatoriai. Vadinasi, net jeigu "keyloggeris" pavogs asmens PIN kodą, jeigu naudosime ir trečiąją saugumo lygį – atskirą fizinį įrenginį (pultelį) – kodų generatorių – tai virusas niekaip nesugebės prabalsuoti papildomai be vartotojo žinios, po to kai jis prabalsuos pirmą kartą ir balsą pasitikrins kitame įrenginyje – telefono ekrane. Taip yra todėl, kad kodu generatorius kaskart sugeneruoja vis kitą atsakymą, ir tik ją žinant įmanoma prisijungti prie sistemos ir prabalsuoti.

5. Rizika – duomenų perėmimas juos persiunčiant iš balsuotojo kompiuterio į serverį

6.1. Rizika ir jos sprendimas

Baimė – *"balsuotojo duomenis "hakeriai" perims juos persiunčiant, ir serverio nepasieks, arba persiunčiant bus pakeisti kitu balsu, arba balsuotojo asmens duomenys bus atskleisti persiunčiant"*

Išvada – duomenys iš kliento kompiuterio į serverį perduodami saugiai HTTPS protokolu, naudojant brangų ir "žalią" (angl. "Extended Validation") aukšto lygio sertifikatą ("Verisign", "Geotrust"), ar net "Root" sertifikatą, skirtą valstybinėms institucijoms, naudojančioms *.gov domeną. Toks sertifikatas yra įtraukiamas į oficialius "Windows" operacinės naujinimo paketus.

5.1. Kaip apsisaugoti nuo balsuotojo duomenų atskleidimo persiunčiant duomenis?

Atsakymas – balsuotojo duomenys turėtų būti užšifruojami dar prieš juos nusiunčiant išsaugojimui. Tai reiškia, kad jie turi būti užšifruojami ne serveryje, o balsuotojo kompiuteryje.

Tai galima padaryti:

- jQuery Front-End kodo bibliotekų pagalba, arba
- Ajax HTTPS $_GET JSON(P) tipo užklausa, kurioje atsakymo į užklausą duomenys būtų gaunami JSON formatu. Kreipiamasi būtų tam tikrą servisą, kuris gautų informaciją tik apie balsuotoją, bet negautų informacijos apie jo balsą.

Tada, antruoju etapu, būtų nusiunčiami balsuotojo ir balso duomenis į internetinio balsavimo sistemos serverį saugojimui duomenų bazėje. Šiame, antrajame etape, būtų persiunčiamos abi reikšmės – tiek užšifruota balsuotojo informacija, tiek neužšifruotas balsas. Jeigu sistema, prieš įvykdydama "INSERT" užklausą, "SELECT" užklausos metu aptinka duomenų bazė informaciją, kad jau tokia šifruota rinkėjo informacija egzistuoja, o nešifruoto balso laukelyje (angl. "vote") aptinka atšaukimo reikšmę (angl. "CANCELED"), tai vadinasi toks klientas jau balsavo apylinkėje įprastu, popieriniu būdu, bei prieš balsavimą popieriniu būdu, apylinkėje esančiu kompiuteriu atšaukė savo balsą.

Alternatyvi "atšaukto balso" realizacija, balsavusiems "popieriniu" būdu, būtų saugoti duomenų bazėje duomenis papildomoje "electorate" lentelėje, kurioje būtų nešifruoti laukeliai su vardu, pavarde, asmens kodu ir balsavimo statusu (balsavo, atšaukė). Pastaruoju atveju atšaukti rinkėjo galimybę balsuoti internetu, galėtų rinkimų komisijos narys rinkimų komisijos kompiuteryje, o pačiam rinkėjui to daryti nereikėtų, nebent jis jau būtų pats anksčiau balsavęs internetu – tokiu atveju atšaukti savo internetinį balsą galėti tik pats rinkėjas, nes tik jis su savo asmens tapatybės kortele, joje esančiu el. parašu ir tik rinkėjui žinomu PIN kodu galėtų

sugeneruoti naują įrašą lentelėje “votes”, su identišką buvusiai užšifruotai balsuotojo informacijos laukelio reikšme, bei laukelio “vote” reikšme “CANCELED”.

5.2. Kaip užtikrinti, kad vartotojo balsas tikrai nebuvo pakeistas jį persiunčiant?

Atsakymas – norint įsitikinti, kad balsas ir vartotojo duomenys buvo nepakeisti, vartotojas turėtų tai patikrinti antrajame įrenginyje – mobiliajame telefone -, kuriame, per 30 sekundžių po balsavimo kompiuteriu, nuskanuotų kompiuterio ekrane pasirodžiusį QR kodą. Mobiliojo telefono programėlė turėtų sugebėti sugeneruoti užšifruotą vartotojo balso reikšmę ir sutikrinti pagal QR kode išsaugotus duomenis su reikšme, esančia serverio duomenų bazėje.

Šiame procese dalyvauja 3 objektai:

- serverio programinė įranga,
- balsuotojo kompiuteryje esanti saugumo reikalavimus atitinkanti interneto naršyklė,
- balsuotojo mobiliajame telefone esant programėlė.

Tokiu atveju duomenys šifruojami dviejuose įrenginiuose – telefone ir balsuotojo kompiuteryje, bei sutikrinami taip pat dviejuose įrenginiuose – serveryje ir balsuotojo telefone.

Balsavimo duomenų nusiuntimo (gavimo) laiką sugeneruos pats serveris UNIX_TIMESTAMP() būdu remdamasis oficialia atominio laikrodžio tarnyba. Informacija apie balsavimo laiką bus pateikiama balsavusiojo asmens telefone balso verifikavimui, o taip pat bus išsaugota duomenų bazės balsų lentelėje, kartu su užšifruota balsuotojo informacija ir balsu.

5.3. Kaip saugiai persiųsti duomenis iš kliento kompiuterio į serverį? (HTTPS)

Atsakymas – duomenys iš kliento kompiuterio į serverį perduodami saugiai HTTPS protokolu, naudojant brangų aukšto lygio sertifikatą, turintį išplėstinio saugumo patvirtinimo EV SSL (angl. “Extended Validation”) žymą. Tokia žyma interneto naršyklėse tinklapio adreso juostoje dažniausiai žymima žalia “spynelės” piktograma, arba pati adreso juosta tampa žalia (žr. :

2 pav. EV SSL sertifikatu sertifikuoti tinklapiai naršyklės juostoje dažniausiai žymimi žaliai

Aukšto lygio "žalius" sertifikatus išduoda tokios kompanijos kaip "Verisign", "Geotrust". Jų kaina prasideda nuo $ 499 USD/metams (pvz. antrasis sertifikatas čia – http://www.geotrust.com/ssl/wildcard-ssl-certificates/).

Verta paminėti, ir dar du "pilkus" ir "mėlynus" sertifikatus:

- pilka "spynelės" piktograma, arba pilka adreso juosta interneto naršyklėje žymimi pasibaigę ar atšaukti (angl. "broken") sertifikatai žymimi .
- mėlyna "spynelės" piktograma, arba mėlyna adreso juosta interneto naršyklėje žymimi pigesni "RapidSSL" tipo sertifikatai

Plačiau – http://support.hostgator.com/articles/ssl-certificates/ssl-setup-use/color-bars-for-ssl .

Valstybinėms institucijoms, E-Valstybės (angl. "E-Government") sistemoms, dažniausiai naudojančioms *.gov domeną, valstybinėms sertifikavimo tarnyboms (angl. "Government CA") būna išduodami ir dar aukštesnio lygio sertifikatai – vadinamieji "sisteminiai sertikatai" (angl. "Root Certificate") su "auksiniu raktu" (angl. "Golden Key"), kuris dažniausiai yra saugomas aparatinės įrangos pagalba (angl. "hardware security module"). Kompanija "Microsoft" iki šiandien dienos patvirtino apie 100 CA tarnybų.

"***Government CAs***

Increasingly national and regional governments are establishing Certification Authorities intended primarily for government to government or citizen to government (e-government) transactions. "

Plačiau rašoma straipsnyje "Microsoft Root Certificate Program" – http://msdn.microsoft.com/en-us/library/cc751157.aspx .

Visi "Root" sertifikatai yra įtraukiamas į oficialius "Windows" operacinės atnaujinimo paketus. Tai reiškia, kad jeigu naudojant paprastą sertifikatą vartotojas naršyklėje dar turi pats paspausti iššokusiame

lange (angl. “pop-up”) , kad pasitiki ir patvirtina šį sertifikatą, tačiau jeigu neturi žinių apie sertifikavimo tarnybą, jis negali žinoti ar gali ja pilnai pasitikėti. Kai tuo tarpu aukščiausio lygio sertifikatai yra patvirtinti pagal nutylėjimą iš operacinės sistemos pusės.

Jeigu yra prisibijoma NSA ir panašių JAV tarnybų, nuo orderio gauti prieigą prie sertifikavimo tarnybos centro, valstybė pati gali sukurti savo sertifikavimo tarnybą, naudodama “auksinį raktą” (angl. “Golden Key”).

Aukšto lygio sertifikatas yra išduodamas ne tik, kaip įrodymas, bet ir įmonė ir jos tinklapis turi atitikti tam tikrus reikalavimus, kad gautų šį sertifikatą:

- Yra tikrinamas tinklapio domenas, sub-domenas, ir tinklapio puslapiai.
- Yra patikrinama ar duomenys duomenų bazėje viduje yra šifruotai saugomi
- Ar darbuotojai laikosi saugumo reikalavimų ir jų kvalifikacija yra pagrįsta reikiamais sertifikatais

“Root” sertifikatams ir juos išduodančioms sertifikavimo tarnyboms yra taikomi dar griežtesni techniniai reikalavimai, pavyzdžiui turi atitikti RSA 2048-bitų ilgio reikalavimą:

> *“4. Root certificates must comply with the Technical Requirements section below. In particular, we require a minimum crypto key size of RSA 2048-bit modulus for any root and all issuing CAs. Microsoft will no longer accept root certificates with RSA 1024-bit modulus of any expiration.”*

“Root” sertifikatai yra labiausiai saugomi HTTPS klasės sertifikatai:

> *“If the Root Certificate, the Golden Key, of a company is stolen, that’s the end of the world, for the company holding it at least. Root CA certificates are among the most valuable things in the hacker underworld. As such, there is usually an extreme level of security inherent in the storage of any device which holds the Golden Key (typically a Hardware Security Module). “*

Plačiau – http://security.stackexchange.com/questions/44787/changing-private-keys .

6. Rizika – balsų pirkimas

6.1. Rizika ir jos sprendimas

Baimė – *"žmogų privers tave balsuoti už norimą kandidatą, arba pirks balsus seansais prie kompiuterio per vakarėlį ar spaudžiamas šeimos nario"*

Išvada – internetinio balsavimo sistemoje būtina realizuoti galimybę balsuoti internetu pakartotinai

6.2. Kaip apsisaugoti nuo balsų pirkimo per renginius?

Atsakymas – šią problemą jau sprendžia Estijos balsavimo internetu sistema. Žmogus balsuoti internetu gali neribotą kiekį skaičių, ir tik jis pats žino, kuris jo balsas bus paskutinis. Todėl, net jeigu ir pirminis balsas buvo "nupirktas per renginį už vyną ir sūrį", žmogus, grįžęs namo, galės prisijungti prie internetinio balsavimo sistemos ir prabalsuoti dar kartą – šį kartą jau už savo norimą kandidatą.

6.3. Kaip reikėtų saugoti duomenis duomenų bazėje, kad rinkėjas galėtų balsuoti pakartotinai tiek internetu, tiek apylinkėje rinkimų dieną?

Atsakymas – pagal "unix_timestamp" (DB lentelėje "votes", kurios kiti trys stulpeliai yra balso eilė, balsuotojo užšifruoti duomenys ir nešifruota informacija apie balsą), į balsavimo apylinkę atėjęs žmogus, galėtų atšaukti savo balsą.

Atšaukus balsą į DB lentelę "votes" būtų įrašyta nauja eilutė su tokia pati sugeneruota šifruota reikšmė apie rinkėją, vietoje balso būtų įrašyta "CANCELED", datos laukelyje – atšaukimo data UNIX formatu.

Alternatyvi "atšaukto balso" realizacija, balsavusiems "popieriniu" būdu, būtų saugoti duomenų bazėje duomenis papildomoje "electorate" lentelėje, kurioje būtų nešifruoti laukeliai su vardu, pavarde, asmens kodu ir balsavimo statusu (balsavo, atšaukė). Pastaruoju atveju atšaukti rinkėjo galimybę balsuoti internetu, galėtų rinkimų komisijos narys rinkimų komisijos kompiuteryje, o pačiam rinkėjui to daryti nereikėtų, nebent jis jau būtų pats anksčiau balsavęs internetu – tokiu atveju atšaukti savo internetinį balsą galėti tik pats rinkėjas, nes tik jis su savo asmens tapatybės kortele, joje esančiu el. parašu ir tik rinkėjui žinomu PIN kodu galėtų sugeneruoti naują įrašą lentelėje "votes", su identišką buvusiai užšifruotai balsuotojo informacijos laukelio reikšme, bei laukelio "vote" reikšme "CANCELED".

6.4. Kokioje situacijoje pakartotinis balsavimas yra neįmanomas?

Tai itin reta situacija, bet ji būtų tokia – jeigu rinkėjo el. parašo sertifikatas baigtų galioti per rinkimų periodą, o jis jau būtų kartą prabalsavęs. Tuomet, pakeitus asmens tapatybės kortelę su el. parašu į naują kortelę, rinkėjas nebegalėtų sugeneruoti buvusios užšifruotos reikšmės. Tokiu atveju rinkėjas galėtų balsuoti tik pirmą kartą, o pakartotinis balsavimas nei internetu, nei apylinkėje būtų neįmanomas.

Šią problemą, kurios tikimybė mažesnė nei 0,1%, esant tokiam poreikiui, galima išspręsti pasirašymui (balso atidavimo momentu) reikalaujant žmogaus kaskart įvesti tarkim 30 skaitmenų saugumo kriterijus atitinkantį slaptažodį (didžiosios, mažosios raidės, skaičiai, specialūs simboliai, tarpai, lietuviškos raidės).

7. Rizika – vienas žmogus prabalsuos už grupę kitų asmenų

7.1. Rizika ir jos sprendimas

Baimė – *„ateis kunigas, ir prabalsuos už visus senus neįgalius parapijiečius taip, kaip jis nori“*

Išvada – visiems, ar tik rizikos grupių, rinkėjams balsuoti internetu turėtų būti privaloma naudojant “**parašo klaviatūros klavišais**” verifikavo sistemą (angl. “**Keystroke Dynamics**“), kurią 2007 metais JAV patentavo jos išradėjai. Iš esmės – kiekvieno žmogus turi unikalų laiko tarpą (ms) tarp skirtingų klaviatūros klavišų kombinacijų paspaudimų, bei unikalų laiką (ms), kurį kiekvienas iš klavišų buvo nuspaustas. Tai 99% tikslumu gali įvardinti faktą, kuomet vienas žmogus balsuoja už kelis asmenis, arba asmuo balsuoja ne pats.

7.2. Kaip neleisti balsuoti už kitą žmogų ir aptikti tą patį žmogų balsuojantį už skirtingus asmenis?

Atsakymas – visiems, ar tik rizikos grupių, rinkėjams balsuoti internetu turėtų būti privaloma naudojant “parašo klaviatūros klavišais” verifikavo sistemą (angl. “**Keystroke Dynamics**“), kurią 2007 metais JAV patentavo jos išradėjai. Iš esmės – kiekvieno žmogus turi unikalų laiko tarpą (ms) tarp skirtingų klaviatūros klavišų kombinacijų paspaudimų, bei unikalų laiką (ms), kurį kiekvienas iš klavišų buvo nuspaustas. Tai 99% tikslumu gali įvardinti faktą, kuomet vienas žmogus balsuoja už kelis asmenis, arba asmuo balsuoja ne pats.

Norint realizuoti šį reikalavimą, kiekvienas rinkėjas, norintis balsuoti internetu su verifikavimo procesu, privalėtų, atsiimdamas asmens tapatybės kortelę (toliau – ATK), įvesti kompiuteryje vadinamuosius klavišų įvesties šablonus (angl. “**Keystroke Patterns**“). Tai iš esmės turėtų būti dviejų įvesties laukų sistema, kur žmogus turėtų įvesti ekrane parodytą rišlaus teksto pastraipą, kurioje būtų panaudota kuo daugiau skirtingų raidžių kombinacijų. Žmogus tą padaryti turėtų du kartus. Tuomet duomenų bazėje būtų išsaugoma mažiausia, didžiausia ir vidutinė reikšmė. Vėliau, sistemos kūrėjai, remdamiesi praktika nurodyti didesnį ar mažesnį sistemos jautrumą vartotojo įvesties validacijai. Šie duomenys turėtų būti saugomi tiek VRK, ar sistemos serveryje, arba ATK serveryje. O taip pat turėtų būti įrašomi į ATK.

Realizuojant šį algoritmą, rinkėjas, balsuodamas kompiuteriu, privalėtų įvesti tiek savo paties vardą, pavardę ir asmens kodą, tiek kandidato, už kurį balsuoja, vardą ir pavardę. Rinkėjo įvestis, paspaudus mygtuką “Atiduoti balsą” būtų patikrinta su ATK esančiais duomenimis, ir aptikus galimą balso klastojimo atveji, turėtų būti kompiuterio ekrane pateiktas pranešimas apie “Netikrą balsą” ir “Draudimą balsuoti internetu” su pasiūlymų tai atlikti balsavimo apylinkėje. Rekomenduotina, kad sistema taip pat tokiu momentu išsiųstų duomenis – rinkėjo vardą, pavardę ir asmens kodą – į VRK tarnybą su rekomendacija susisiekti ir patikrinti rinkėją dėl galimo pažeidimo.

Šią sistemą praktikoje realizuoja "**Coursera**" mokymosi internetu sistema, kuomet siekiama atskirti ar studentas, nenusirašinėja egzamino internetu metu. Žvelgiant plačiau – "Coursera" žengė dar toliau, ir panaudojo net ir internetinės kameros (angl. "**Web-cam**") funkcijas, bei "erdvės objektų sekimo" (angl. "**field tracking**") sistemas, kurios esant visiškai paranojai, galėtų būti realizuojamos ir balsavime internetu.

JAV 2007 metais patentuoto **"Keystroke Dynamics" išradimo pilnas aprašymas**: http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO%2Fsrchnum.htm&r=1&f=G&l=50&s1=7206938.PN.&OS=PN/7206938&RS=PN/7206938

Apie "Keystroke Dynamics" naudojimo "Coursera" studijų internetu sistemoje rašoma šiame Washington Post" straipsnyje: http://www.washingtonpost.com/blogs/college-inc/post/moocs–here-come-the-credentials/2013/01/09/a1db85a2-5a67-11e2-88d0-c4cf65c3ad15_blog.html

Apie "Keystroke Dynamics" sistemą ir "Keystroke logging" algoritmus rašoma ir "Wikipedia" enciklopedijoje:
http://en.wikipedia.org/wiki/Keystroke_dynamics
http://en.wikipedia.org/wiki/Keystroke_logging

8. Rizika – rinkimų metu sistema bus nepasiekiama dėl DDoS atakų

8.1. Rizika ir jos sprendimas

Baimė – *„ 'hakeriai' užlauš sistemą DDoS atakomis ir padarys ją rinkimų metų nepasiekiama visiems ar daliai gyventojų.“*

Išvada – sistema turi būti realizuota kaip decentralizuota, turi būti pasirinkti saugūs architektūriniai sprendimai, turintys aukšto lygio ugniasienes (angl. “Firewall”) ir srauto balansavimo (angl. “Load Balancer”) sistemas, esančias priekyje programos serverių. Arba turi būti pasirinkti debesų kompiuterijos (angl. “Cloud”) sprendimai, kaip “Microsoft Azure”, “Amazon AWS” ar “Google Cloud”, jau realizuojantys šias apsaugas. Serveriai turi būti įskirstyti skirtingų IP adresų tiekėjų tinkluose. Sistemos veikime turėtų būti numatyti skirtingi darbo ir apkrovos režimai su skirtingu jautrumu “DDoS” atakoms.

8.2. Kaip apsaugoti sistemą nuo DDoS atakų?

Nuo paskirstyto prieigos prie sistemos blokavimo – DDoS atakos (angl. “distributed denial-of-service”), kuomet tūkstančiai ar milijonai tam tikrame regione ar visame pasaulyje užkrėstų kompiuterių, sujungtų į atakuotojo valdomą “botnet” virusuotų kompiuterių tinklą, vykdo tikslingas užklausas į tam tikrus tinklapius ar serverių IP adresus, turėtų būti saugomasi taip:

1. Sistema turėtų būti decentralizuota. Tai reiškia, kad sistemos ir jos duomenys turėtų būti saugomi 10-yje skirtingų serverių, esančių skirtinguose vietose, prijungtų skirtingais IP adresais. Šis sprendimas taip pat minimizuoja duomenų pakeitimo riziką (balso suklastojimą).
2. Turėtų būti naudojamasi patikimais architektūriniais sprendimais, arba turėtų būti pasirinkti sprendimai, realizuojantys tokią apsaugą – tokias apsaugas siūlo debesų kompiuterijos (angl. “Cloud”) paslaugų tiekėjai – “Microsoft Azure”, “Amazon AWS”, “Google Cloud” ir kiti.
3. Turėtų būti naudojamos itin aukšto lygio ugniasienės (angl. “firewall”) – atskiri serveriai, peržiūrintys kiekvieną kreipinį dar iki jam patenkant iki galutinį – HTTP Request užklausos apdorojimo serverio, puikiai filtruojančios kenksmingas užklausas iš virusais užkrėstų kompiuterių, ir jas blokuojančios patalpindamos jas į blokuotų IP adresų lenteles serveryje (angl. “IP tables”). Šias apsaugas realizuoja žinomi “Cloud” paslaugų tiekėjai – “Microsoft Azure”, “Amazon AWS”, “Google Cloud” ir kiti.
4. Kiekviena iš naudojamų sistemos instancijų, tarkim jeigu tokių instancijų yra 10, turėtų būti paremta hierarchine architektūra. T.y. prieš duomenų apdorojimo serverį, turėtų būti apkrovos paskirstymo (angl. “load balancer”) serveris, kuris, gali būti tas pats arba kitas nei ugniasienės (angl. “firewall”) serveris. “Load balancer” serveris ne tik tolygiai paskirstytų užklausas įvairiems serveriams, bet kartu ir stebėtų užklausų kiekio augimą, ir pastebėjęs, kad iš tam tikro miesto, regiono, šalies, ar apskritai tam tikru laiko momentu neįprastai greitai pradėjo daugėti HTTP užklausų (angl. “HTTP Request”),

tokios perteklinės užklausos turėtų būti traktuojamos kaip potenciali rizika sistemos darbui, bei blokuotos – patalpintos į blokuojamų IP adresų lenteles (angl. “IP tables”) visam laikui arba laikinai tam tikram periodui.

5. Vykstant pasaulinei DDoS atakai rekomenduotinas sprendimas būtų laikinai blokuota prieigą prie sistemos iš visų IP adresų, kurie nepriklauso interneto tiekėjams Lietuvoje. Tai būtų aukščiausias apsaugos lygmuo veikiančiam serveriui, ir galėtų būti taikomas tik esant multi-milijoninei atakai.
6. Griežčiausias sprendimas, kurio teorinė galimybė yra mažiau nei 0,01%, yra sustabdyti serverių darbą iki ataka pasibaigs. Tokiu atveju atakai nepasibaigus rinkimų dieną, rinkėjas balsuotų rinkimų apylinkėje. Su tokio masto ilgalaike ataka – trunkančia ilgiau nei keliolika valandų – pasaulio kompiuterių istorijoje – neteko susidurti dar nei vienai kompiuterinei sistemai, dėl elementarios priežasties – užkrėstų kompiuterių IP adresų kiekis, blokuojant vis naujus adresus, pasibaigtų per 10-15 valandų, net ir sujungus daugumą didžiausių pasaulyje “botnet’ų” į vieną tinklą.

Plačiau – http://web.archive.org/web/20100914222536/http://anml.iu.edu/ddos/types.html

9. Rizika – negalėsime pasitikėti sukurta sistema

9.1. Rizika ir jos sprendimas

Baimė – *"sistemą sukurs nekvalifikuoti papirkti specialistai, ir sistema bus padaryta ne taip kaip sakoma, ir niekas negalės to patikrinti"*

Pirmoji išvada – tam tikros sistemos dalys turėtų būti paviešintos atviro kodo principu viešoje kodo saugykloje, pavyzdžiui "GitHub". Paviešintos turėtų būti sistemos dalys, kuriose vyksta duomenų saugojimas duomenų bazėje, duomenų užšifravimas, bei tos kuriose tikrinamas duomenų autentiškumas.

Antroji išvada – po sistemos sukūrimo turi būti atliktas sistemos auditas, kurį turi atlikti ne mažiau kaip dvi skirtingos kvalifikacijos reikalavimus atitinkančių specialistų turinčios agentūros ar įmonės. Tokie kvalifikaciniai reikalavimai galėtų būti kaip aukštasis universitetinis bakalauro, magistro ar mokslų daktaro išsilavinimas IT srityje, tarptautinis CISCO CNNA, CCNP ar CCIE sertifikatas, "Coursera" sistemoje pabaigto **"**Securing Digital Democracy" kurso pažymėjimas, arba 5 metų patirtis dirbant su IT saugumu susijusiuose projektuose.

9.2. Kaip sukurti internetinio balsavimo sistemą, kuria galima pasitikėti?

Sistema turėtų būti sukurta taip, kad tam tikros sistemos kertinės dalys, kuriose vykdomas duomenų šifravimas, autentiškumo tikrinimas ir išsaugojimas duomenų bazėje, būtų paviešintos atviru kodu. Plačiau apie kokias konkrečias sistemos dalis reikėtų paviešinti atsakyta kitame klausime.

Taip pat sukūrus sistemą, turėtų būti privalomą atlikti jos auditą. Jį turėtų atlikti bent dvi skirtingos kvalifikuotų IT saugumo specialistų turinčios įmonės. Detalizuoti reikalavimai tokių specialistų kvalifikacijai nurodyti trečiame šio skyriaus klausime.

9.3. Kokios internetinio balsavimo sistemos dalys turi būti paviešintos atviru kodu?

Atviro kodo turėtų būti tos sistemos kodo dalys, kuriose apibrėžiamas duomenų užšifravimas ir išsaugojimas duomenų bazėje:

- Kad kiekvienas kvalifikuotas specialistas galėtų įsitikinti, kad balsuotojo asmens duomenys iš tiesų yra užšifruojami, ir saugomi užšifruotų formatu, ir toje pačioje duomenų bazės lentelėje nėra išsaugotų neužšifruotų jo asmens duomenų, nešifruotai susisiejančių jį su konkrečiu balsu.
- Kad kiekvienas kvalifikuotas specialistas galėtų įsitikinti, kad jeigu ir yra saugomi nešifruoti asmens vardas, pavardė ir asmens kodas, jie būtų saugomi atskiro duomenų bazės lentelėje, kurioje saugoma informaciją apie patį balsavimo faktą, o ne konkretų balsą.

- Jeigu duomenys yra šifruojami vartotojo kompiuteryje Front-End jQuery/Javascript/Ajax/JSON(P) skriptų ir metodų pagalbą, toks Front-End kodas taip pat turėtų būti viešai prieinamas peržiūrai viešoje kodo saugykloje, pavyzdžiui "GitHub".

Panašias į aukščiau paminėtas praktikas taiko estiškos balsavimo internetu sistemos kūrėjai.

Taip pat turėtų būti galimybė kiekvienam kvalifikuotam specialistam peržvelgti sistemoje, kaip yra panaudojamas ORM modelis užklausų į duomenų bazę formavimui, kaip formuojamos transakcijos, kaip inicijuojama jų pradžia ir pabaiga, bei apsisaugoma nuo duomenų klastojimo tikrinant duomenų autentiškumą.

9.4. Kokie turėtų būti keliami reikalavimai sistemos auditui pabaigus ją kurti?

Sistemą turėtų audituoti kvalifikuoti IT specialistai:

- arba turintys aukštąjį universitetinį išsilavinimą šioje srityje, pirmenybę teikiant magistro ar mokslų daktaro laipsnį turintiems specialistams
- arba sertifikuoti specialistai, turintys bent CISCO CCNA tarptautinį sertifikatą, o pirmenybę teikiant turintiems CISCO CCNP ar CISCO CCIE sertifikatą ir turintys aukštąjį universitetinį išsilavinimą šioje srityje
- arba išklausę ir išlaikę "Coursera" kurso "Securing Digital democracy" kursą ir turintys aukštąjį universitetinį išsilavinimą šioje srityje
- arba ekspertai, dirbantys IT saugumo srityje bent 5 paskutinius metus, bei turintys aukštąjį universitetinį išsilavinimą šioje srityje

Sistemą audituotų turėtų daugiau nei vieną, t.y. bent dvi, o rekomenduotina – trys, skirtingos įmonės ar audito agentūros, kuriose dirba anksčiau apibrėžtus kvalifikacijos reikalavimus atitinkantys specialistai.

10.Rizika – nežinomi programiniai sprendimai

10.1. Rizika ir jos sprendimas

Baimė – *"bus naudojami niekam nežinomi programiniai sprendimai, kurie praktikoje neegzistuoja"*

Išvada – yra gausybė įvairių tinkamų programinių sprendimų, kuriuos galima rinktis arba pagal jų populiarumą Lietuvoje, arba pagal įmonės, laimėsiančios viešąjį konkursą, specialistų įgūdžius.

10.2. Kokią programavimo kalbą naudoti e-balsavimo interneto svetainei?

Atsakymas – saugią sistemą galima padaryti su bet kuria populiaria programavimo kalba, tinkama naudoti asmeniniuose kompiuteriuose – Php, C# .NET, Python, Ruby on Rails, Java EE, Objective C, C++ ar C.

10.3. Kokią duomenų bazę naudoti e-balsavimo duomenims saugoti?

Atsakymas – saugią sistemą galima padaryti su bet kuria populiaria SQL reliacine duomenų bazių valdymo sistema (RDBVS), objektine duomenų baze ar NoSQL (angl. "Not only SQL") duomenų bazė. Todėl tinka nemokama MySQL, mokama Oracle DB, PostgreSQL, NoSQL dokumentų bazės MongoDB, Microsoft SQL Server, IBM DB2.

10.4. Kokią programinę įrangą (OS) naudoti serveriuose?

Atsakymas – serveriuose rekomenduoju naudoti bent dvi skirtingų šeimų operacines sistemas. Geriausia – vieną iš "Windows", o kitą iš OS "Linux" sistemų šeimos. Pageidaujant trečioji OS galėtų būtų vieną iš kitų "Unix" OS sistemų grupės.

10.5. Kokie reikalavimai turėtų būti keliami prisijungimui prie e-balsavimo sistemos administravimo?

Atsakymas – dirbant administravimo režimu, serveryje turėtų būti parinkti nustatymai taip, kad prie serverio su administracine sistemos dalimi galėtų prisijungti tik tam tikri kompiuteriai su į "saugu sąrašą" patenkančių parametrų visuma:

- IP adresu,
- MAC adresu,

- ekrano rezoliucija,
- interneto naršyklės pavadinimu ir versija,

Kuriant balsavimo internetu sistemą, turėtų būti taikomas sistemų atskyrimo principas didesniam sistemos saugumui. Toks principas sako, kad sistemos versija su administruojamąją dalimi turėtų būtų neprieinama rinkimų laikotarpiu. Prie sistemos su administruojamąja dalimi būtų galima prisijungti tik pasibaigus rinkimų laikotarpiui, padarius esamų balsų duomenų bazių kopijas ir padėjus jų kelias kopijas jas į atskiras, internetu neprieinamas duomenų saugyklas.

Tuomet prie balsavimo internetu sistemos su administravimo dalimi būtų prisijungiama vidiniame kompiuterių tinkle arba tik iš tam tikro IP adreso, MAC adreso, su tam tikra rezoliucija, interneto naršyklė ir jos versija, kartu būtų naudojama naujausia duomenų bazės kopija. Taip būtų suskaičiuojami balsai.

10.6. Kokie reikalavimai turėtų būti keliami duomenų išsaugojimui duomenų bazėje?

Atsakymas – balsuojant rinkėjui, tarkim, MySQL duomenų bazėje užklausą realizuojančio "MySQL user" sistemos kliento teisės turėtų būti tik "SELECT" ir "INSERT". Saugumo tikslais "GRANT", "DROP", "DELETE", "UPDATE" ir kitos nesaugios užklausų komandos turėtų būti draudžiamos. Taip būtų užtikrinamas sistemos duomenų pakeitimų atsekamumas net ir juos pakeitus, nes kaskart įvykdžius pokyti sistemoje, duomenų bazės lentelėje būtų įterpiama vis nauja eilutė, o skaičiuojant balsus būtų laikomasi principo kad įskaičiuojamas tik tas balsas, kuris turi vėliausią (didžiausią) išsaugotą UNIX_TIMESTAMP laiko reikšmę tam balsuotojui, remiantis šifruotais rinkėjo duomenimis (angl. "hash").

Iš programavimo kalbos kodo pusės, užklausos duomenų bazėje neturėtų būti vykdomos tiesiogiai:

1. Užklausos iš kodo pusės turėtų būti vykdomos ORM modelio (angl. "object-relation model") pagalba, jeigu būtų naudojama reliacinė DBVS. Taip esama sistema nebūtų susieta su specifine DBVS, ir ateityje galėtų būti pakeista. Toks programinis kodas būtų nuo DBVS nepriklausomas (angl. "platform independant"). Arba turėtų būti naudojama objektinė DBVS. Naudojant ORM modelį, po ORM įvykdytos užklausos kodo transformacijos gautas RDBVS užklausos sakinys turėtų būti supranta kuo platesniam programuotojų ratui.
2. Duomenų bazė privalo palaikyti transakcijas. Jas palaiko, pavyzdžius InnoDB saugomo variklis (angl. "storage engine"), naudojamas MySQL DBVS. Transakcijos turėtų būti naudojamos saugant duomenis. Tai ypač aktualu, jeigu įrašas yra saugomas dviejose lentelėse su šifruotais ir nešifruotais duomenimis, ir siekdami didesnio saugumo, šias užklausas dvi susijusias vykdysime atskiromis dviem atskiromis operacijomis.

10.7. Kokią programavimo kalbą būtų praktiškiausia pasirinkti Lietuvoje?

Atsakymas – Lietuvoje naudingiausia būtų pasirinkti **Php** programavimo kalbą, nes ją supranta daugiausia žmonių Lietuvoje ir užsienio lietuvių – Lietuvos ir užsienio lietuvių Php programuotojų bendruomenę šiuo metu sudaro daugiau nei 25.000 narių. Kai tuo tarpu kitų programavimo kalbų paplitimas Lietuvoje yra bent 5-10 kartų mažesnis. Todėl, jeigu siekiame tam tikras sistemos dalis paviešinti atviru kodu ir įdėti į jas viešąją kodo saugyklą internete (pvz. "GitHub"), tai kiekvienas iš šių 25.000 žmonių būtų potencialus auditorius. Kitų programavimo kalbų atveju tokių potencialių auditorių turėtume tik 1-5 tūkstančius.

Svarbiausias, kad būtų pasirinkta ta programinė kalba, kurią geriausiai mokės tos įmonės specialistai, kuri laimės viešąjį konkursą kurti ar diegti programinę įrangą VRK serveriuose, duomenų centruose ar debesų serveriuose (angl. "Cloud Servers"). Arba kitu atveju programinė kalba turėtų būti nurodyta viešajame konkurse, ir pasirinkta tokia įmonė, kuri turi labiausiai patyrusius šios programavimo kalbos specialistus.

10.8. Kokią DBVS būtų praktiškiausia pasirinkti Lietuvoje?

Atsakymas – Lietuvoje naudingiausia būtų pasirinkti reliacinę **MySQL DBVS**, nes jos SQL sakinių sintaksę supranta per 14.000 programuotojų Lietuvoje ir užsienio lietuvių tarpe. Kai tuo tarpu NoSQL MongoDB, Microsoft SQL ir kitų duomenų bazių užklausų-atsakymų sintaksę, prasmę, struktūrą ir saugumą supranta vos 1-2 tūkstančiai programuotojų Lietuvoje. Jeigu būtų pasirinkta ši reliacinė DBVS, programinio kodo dalyje neturėtų būti rašomos SQL užklausos tiesiogiai, bet naudojamas ORM modelis (angl. "object-relation model"), kuris sistemos programinį kodą paverstų nuo DBVS nepriklausoma platforma (angl. "platform independant"). Tačiau po ORM modelio operacijos į MySQL užklausos kodą transformuota užklausa, turėtų būti supranta kuo platesniam programuotojų ratui. MySQL DBVS taip pat turi duomenų transakcijas palaikantį saugojimo variklį – InnoDB, kuris ir turėtų būti naudojamas duomenų saugojimui.

Plačiau – http://dev.mysql.com/doc/refman/5.1/en/storage-engines.html .

Svarbiausia, kad yra būtų pasirinkta tokia duomenų bazę, kurią geriausiai mokės tos įmonės specialistai, kuri laimės viešąjį konkursą kurti ar diegti programinę įrangą VRK serveriuose, duomenų centruose ar debesų serveriuose (angl. "Cloud Servers"). Arba kitu atveju programinė kalba turėtų būti nurodyta viešajame konkurse, ir pasirinkta tokia įmonė, kuri turi labiausiai patyrusius šios programavimo kalbos specialistus.

10.9. Kokią serverio OS būtų praktiškiausia pasirinkti Lietuvoje?

Atsakymas – šiuo metu viena populiariausių, bei lengviausiai suprantama, ir lengvai atnaujinama serverių operacinė sistema yra “Ubuntu Server”. Ji turi ir “LTS” ilgo palaikymo versijas (“Long-term support”). Ubuntu LTS OS versijas išleidžia kas du metus, ir naujausioji šiuo metu – Ubuntu 14.04.1 LTS – buvo išleista su 5 metų palaikymu. Šią OS galima pasirinkti Amazon AWS serveriuose. Alternatyvi panaši OS yra “Amazon Linux”.

Antroji populiari platforma, palaikoma “Windows Azure” debesų serveriuose, yra “Windows Server” šeimos operacinė sistema. Naujausias tokia šiuo metu yra – “Windows Server 2012 R2″.

Iš kitų “Unix” OS sistemų grupės šiuo metu galėtų būti “FreeBSD 9.X” OS, “Solaris 11″ ar “Mac OS X 10.10 (Yosemite Server 4.0)”.

11.Mitas – sistema naudojasi tik estai

Mitas – *“pasaulyje internetu balsuoja vieninteliai estai”*

Netiesa. Daugiau nei 30 šalių išbandė balsavimo internetu sistemą vienokia ar kitokia jos formą, skirtingo pobūdžio rinkimuose. O šiuo metu didžiausią rinkėjų internetu bazę turi net ne Estija, o Kanada. Skirtumas tik tas, jog kol kas tik estai vieninteliai ją naudoja taip plačiai – įtraukdami ir Prezidento bei Parlamento rinkimus.

12.Mitas – norvegų modelis buvo daug geresnis nei estų

Mitas – *"lietuviai nupirks 'blogą' estišką modelį, o daug geresnio Norvegiško modelio buvo atsisakyta"*

Netiesa. Norvegiškas modelis nėra geresnis už estiška. Iš tiesų jis daug blogesnis, nes jis nesprendžia esminių balsavimo problemų – nei saugumo, nei galimybės pritraukti daugiau rinkėjų balsuoti, ypač užsienio emigrantų.

Taip yra todėl, kad:

1. Norvegiškame balsavime unikalius balsavimo kodus, kurių neturėtų žinoti niekas, žmonėms išnešioja ir išsiuntinėja paprasti pašto tarnybų kurjeriai. Internetinio balsavimo atveju, jeigu žmogus nusprendžia balsuoti internetu, neturėtų vykti jokie "popieriniai procesai". Nes papirktas kurjeris gali nuskanuoti visus turimus rinkėjų kodus ir vėliau, to pagalba dešifruoti balsus. Unikalų kodą turi žinoti tik pats balsuotojas, jis pats turi jį sugalvoti.
2. Norvegiškas balsavimas yra visiškai nepraktiškai – jis paremtas tuo, kad kiekvienas gyventojas turi būti deklaravęs savo fizinį adresą, ir jis turi būti nepasikeitęs. Kai tuo tarpu ypač užsienio lietuvių atveju, gyvenant kitoje šalyje, adresas gali keistis kad ir kas pora mėnesių, o dažnu atveju adresas apskritai skiriasi nuo gyvenusios vietos, kai žmonės gauna laiškus į darbą, ar į specialias pašto dėžutes, kurioms adresas sukuriamas ir galioja tik 7-10 dienų. Tad Norvegišku modeliu pasinaudoti praktiškai beveik neįmanoma gyvenant emigracijoje.

Tačiau, būtina pabrėžti, kad norvegiškas modelis turi saugumo ir patikimumo privalumų prieš estišką, dėl dalinai decentralizuotos sistemos naudojimo ir didesnio jos atvirumo balsų patikrinimui.

Todėl geriausia internetinio balsavimo sistema būtų tokia:

- ji apjungtų privalumus iš abiejų – estiškos ir norvegiškos – sistemų
- ji būtų paremta estiškos sistemos pagrindu
- joje būtų realizuoti kiti, šiame straipsnyje paminėti, rekomenduojami saugios balsavimo internetu sistemos reikalavimai.

13.Mitas – e-balsavimas nesukuria jokios naudos valstybei, išskyrus reklamą

Mitas – *"balsavimas internetu nesukuria jokios naudos, jis tiko tik valstybės reklamai, bet ir ten jau pavėlavome"*

Netiesa. Pasak Prezidentės Dalios Grybauskaitės – **tik 14.000 iš 700.000 užsienyje gyvenančių lietuvių balsavo per rinkimus**. Vietoje dviejų laiškų siuntimo, tikimybės, kad balsas pavėluos, bus sukurta saugi, veikianti iškart sistema. O ilgalaikės Estijos perspektyvos rodo, kad po 5 metų naudojimosi, internetu balsuoja jau net trečdalis piliečių. Tad rinkėjų aktyvumas tikrai padidėja.

Taip pat taupomas valstybės biudžetas – "popierinis" referendumas dėl žemės valstybei kainavo net 4 milijonus eurų, į kurį atėjo tik 250.000 žmonių. Egzistuojant internetinei balsavimo sistemai, ilgalaikėje perspektyvoje tokio referendumo kaina būtų buvusi 4 kartus mažesnė – vos 1 milijonas eurų. O be to būtų išvengiama eilių rinkimų apylinkėse, nes žmonių apkrova pasiskirstytų tolygiai, būtų tausojama gamta – žmonės neprivalėtų savo transportu vykti iki balsavimo apylinkės, o balsuotų internetu. Tad tausodami gamtą Lietuvos valstybei teigiamą reklamą pasaulyje darysime dar ilgai.

REZULTATAI IR IŠVADOS

Balsavimas internetu yra saugus tinkamai paruoštoje sistemoje, o jo rizikos yra ne didesnės nei paprasto popierinio balsavimo. Išanalizavus balsavimo internetu kritikų pateiktus nuogąstavimus, padarytos išvados, kad keliamos balsavimo įnternetu rizikos, dėl kurių, kaip teigiama, tokios sistemos realizuoti negalima, **yra nepagrįstos ir suvaldomos**.

SAVOKŲ APIBRĖŽIMAI

Debesų serveris (angl. Cloud Servers) – tai toks virtualus serveris, kurio lokacija gali būti lengvai pakeista, o jo našumas greitai padidintas automatiškai, remiantis serverio apkrovimo parametrais.

Išmanusis telefonas (angl. smartphone) – tai mobilusis telefonas su operacine sistema.

Kompiuterinis klavišų parašas (ang. Keystroke Dynamics) – tai būdas nustatyti asmens tapatybę ir duomenų autentiškumą klaviatūros pagalba. Taip yra todėl, kad kievienas kompiuterio naudotojas tarp skirtingų klavišų paspaudimų užtrukta skirtingą laiką (ms), o taip pat laiko nuspaudęs kiekvieną klavišą skirtingą unikalų laiko tarpą (ms). Norint realizuoti šią sistemą, pirmiausia turi būti turimi duomenys apie vartotojo „įvesties klaviatūra įpročius“ (angl. Keystroke Patterns).

Keylogeris (angl. KeyLogger, Keystroke logging) – tai kompiuterio naudotojo paspaustų klavišų informacijos išsaugojimas. Dažnai santrumpa „Keylogeris“ yra vartojama neigiamame, šnipinėjimo, kontekse, o ilgoji frazės versija dažniau naudojama teigiamame, ar moksliniame kontekse, susijusiame su “kompiuteriniu klavišų parašo” technologija.

Kvantinis kompiuteris – hipotetinė skaičiavimų mašina, galinti atlikti skaičiavimus, pasitelkdama fizikines sistemas, kurioms galioja kvantinio susiejimo ir kvantinės superpozicijos dėsniai. Klasikiniuose kompiuteriuose informacija yra operuojama diskrečiais bitais, o kvantiniuose kompiuteriuose – tolygiai kintančiomis kvantinėmis būsenomis – kubitais. Manoma, kad sukūrus kvantinį kompiuterį su pakankamai daug kubitų (1000) taptų įmanoma išspręsti sudėtingus uždavinius, kurių sprendimas klasikiniuose kompiuteriuose užtruktų milijonus metų.

Operacinė sistema (OS) – speciali programinė įranga, užtikrinanti vartotojo sąsają ir mobiliojo telefono techninės įrangos, taikomųjų programų bei duomenų valdymą.

ORM modelis (angl. object-relation model) – modelis, kuris sistemos programinį kodą paverstų nuo DBVS nepriklausoma platforma (angl. platform independant).

Rakto generavimo užlėtinimas (angl. key-stretching) – tai tokia techologija, kuria remiantis siekiama slaptažodžio maišos rakto (angl. hash) generavimo procesą kaip įmanoma užlėtinti, naudojant daugybę iteracijų, tarkim 9,653. Tokiu būdu sistemos slaptažodis apsaugomas tiek nuo greito dekodavimo, tiek nuo sugeneruotų slaptažodžių lentelių (angl. Rainbow Tables).

RSA kriptografija – kriptografinė Sistema, paramta viešu ir privačiu raktu. Viešą raktą gali gauti visi, o privatųjų turi tik jo saugotojas. Taip duomenys yra užšifruojami viešu raktu, o atšifruojami privačiu raktu.

Slaptažodžių lentelė (angl. Rainbow Table) – tai tokia duomenų bazė - failas, įprastai nuo kelių šimtų gigabaitų iki keliolikos teraibaitų dydžio, kurioje naudojant super-kompiuterį buvo sugeneruotos visos įmanomos reikšmės tam tikroje ženklų ir ilgio grupėje. Turint tokią lentelę galima sužinoti pirminę vienkrypčiu slaptažodžio algoritmu užkoduota reikšmę. Tai nėra 100% garantija, tačiau tikimybė, kad egzistuoja alternatyvus žodis su tokiu pačiu maišos raktu, yra labai maža.

SSL – tai standartinė technologija, skirta saugiai sujungti kliento kompiuterį – interneto naršyklę, el. paštą – su serveriu, į kurį kreipiamasi.

Super-kompiuteris – tai itin greitas, paprastai kambario dydžio kompiuteris, galintis atliktis tūkstančius kartų daugiau skaičiavimų per tą patį laiko tarpą, lyginant su paprastu nešiojamuoju ar staliniu kompiuteriu.

UNIX laikas – laikas UNIX sistemos formatu, išgaunamas UNIX_TIMESTAMP() ar kita specializuota komanda. Tai sistemos laikas sekundėmis nuo 1970 metų sausio 1 dienos. Nuo tada yra skaičiuojamas UNIX laikas visose pasaulio kompiuterinėse sistemose. Šis laikas yra nuolat sutikrinamas su pasaulinėmis laiko tarnybomis, besinaudojančiomis tiksliausiais pasaulyje – atominiais laikrodžiais (angl. atomic watch).

SANTRUMPOS

API – aplikacijų programinė sąsaja (angl. Application Programmable Interface)

GUI – grafinė vartotojo sąsaja (angl. Graphic User Interface)

IDE – integruota kūrimo aplinka (pvz. NetBeans) (angl. Integrated Developer Environment)

DBVS – duomenų bazių valdymo sistema (angl. DBMS)

EV SSL – išplėsto galiojimo saugos prievados lygis (angl. Extended Validation Secure Sockets Layer)

HTTP – hiperteksto persiuntimo protokolas (angl. Hypertext Transfer Protocol)

HTTPS – saugus hiperteksto persiuntimo protokolas (angl. Hypertext Transfer Protocol Secure)

J2ME – Java Platform, Micro Edition

J2SE – Java Platform, Standart Edition

J2EE – Java Platform, Enterprise Edition

JDK – Java kūrėjo paketas (angl. Java Developer Kit)

JSON – Javascript kalbos objektų duomenų žymėjimo tipas (angl. JavaScript Object Notation)

JSONP – JSON objekto tipas su atgaliniu verifikavimu (angl. JSON with padding)

JRE – Java paleisties aplinka (angl. Java Runtime Environment)

JVM – Java virtuali mašina, naudojamas kompanijos „Oracle“ (angl. Java Virtual Machine)

MOOC – Masinis atviras mokslo kursas internetu (angl. Massive Open Online Course)

MOS – moderni operacinė sistema (angl. Modern Operating System)

MVC – modelių-vaizdų valdiklis (angl. Model View controler)

NSA – JAV nacionalinio saugumo agentūra (angl. National Security Agency)

OS – operacinė sistema (angl. Operating System)

ORM – objekto-esybių modelis (angl. object-relation model)

OTASL – PĮ atnaujinimai nuotoliniu būdu (angl. Over the air software loading)

PBKDF2 – rakto pagal slaptažodį išvedimo funkcija (angl. Password-Based Key Derivation Function 2)

PĮ – programinė įranga (angl. Software)

RDBMS – reliacinė duomenų bazių valdymo sistema (angl. Relational Database Management System)

RDBVS – reliacinė duomenų bazių valdymo sistema (angl. RDBMS)

RSA – viešo ir privataus rakto kriptografinė sistema (R. **R**ivest, A. **S**hamir, L. **A**dleman)

SDK – PĮ kūrėjo paketas (angl. Software Developer Kit)

SHA256 – Saugus maišos algoritmas (angl. Secure Hash Algorithm)

UAC - vartotojo paskyros teisių kontrolė (angl. User Account Control).

UI – vartotojo sąsaja (angl. Graphic User Interface)

VM – Virtuali mašina (angl. Virtual Machine)

ŠALTINIAI

Knygos, konferencijų medžiaga, dokumentacijos:

[BSP07] Bender; S. Steven, Postley, J. Howard. Key sequence rhythm recognition system and method. 2007 balandis.
[žiūrėta 2015-01]. Prieiga per internetą:
<http://patft.uspto.gov/netacgi/nph-Parser?Sect1=PTO1&Sect2=HITOFF&d=PALL&p=1&u=%2Fnetahtml%2FPTO%2Fsrchnum.htm&r=1&f=G&l=50&s1=7206938.PN.&OS=PN/7206938&RS=PN/7206938>

Elektroniniai teikiniai:

[Geo15] GeoTrust. SSL Certificates: True BusinessID Wildcard
[žiūrėta 2015-01]. Prieiga per internetą:
http://www.geotrust.com/ssl/wildcard-ssl-certificates/

[Hos15] Hostgator.com. Color bars for SSL
[žiūrėta 2015-01]. Prieiga per internetą:
http://support.hostgator.com/articles/ssl-certificates/ssl-setup-use/color-bars-for-ssl>

[Mic15] Microsoft, Inc. Microsoft Root Certificate Program
[žiūrėta 2015-01]. Prieiga per internetą:
<http://msdn.microsoft.com/en-us/library/cc751157.aspx>

[Mys15] MySQL 5.1 Reference Manual. Table 14.1 Storage Engine Features
[žiūrėta 2015-01]. Prieiga per internetą:
<http://dev.mysql.com/doc/refman/5.1/en/storage-engines.html>

[Was13] Washington Post. MOOCS — Here come the credentials. 2013 sausis.
[žiūrėta 2015-01]. Prieiga per internetą:
<http://www.washingtonpost.com/blogs/college-inc/post/moocs–here-come-the-credentials/2013/01/09/a1db85a2-5a67-11e2-88d0-c4cf65c3ad15_blog.html>

[Wik15] Neoficialus šaltinis / Wikipedia. Keystroke dynamics
[žiūrėta 2015-01]. Prieiga per internetą:
<http://en.wikipedia.org/wiki/Keystroke_dynamics>

[Wik15] Neoficialus šaltinis / Wikipedia. Keystroke logging [žiūrėta 2015-01]. Prieiga per internetą: <http://en.wikipedia.org/wiki/Keystroke_logging>