Election Audit Sampling Plan Design— It's Not Just About Sampling Without Replacement

By
Jerry Lobdill
October 9, 2006

Introduction

The design of a sampling plan for an election audit depends on precinct vote count distribution and the assumptions made about the attacker's motivations, risk averseness, desire to succeed, and ability to attack (wholesale vs retail). It is also influenced by the stated purpose of the audit. This paper will discuss these factors, and a sampling plan will be defined. The purpose of the sampling plan defined here is to detect tampering with a 0.99 probability with the greatest possible efficiency.

In this paper it will be assumed that the population of auditable entities are the precincts involved in the race in question and that the race is a two-candidate race.

The Attack

Some researchers have apparently felt that any assumption made about the attack would incur risk of failure of the audit. These researchers do not discuss either the attacker or the attack, but proceed immediately to calculate the sample size to be selected at random from the total population of precincts involved in the election and, perhaps, a particular number of corrupt precincts assumed to be scattered among the entire population at random[1]. They prescribe that once a random sample of precincts has been selected the auditing will proceed until either the last selected precinct has been audited or a corrupted precinct has been found. The sample is presumably audited in the random order in which the list is prepared since the subject is never discussed.

Though proponents of this approach recoil at the thought of making any assumptions about the potential attack, this approach does not completely avoid matters of judgment since the number of corrupt precincts is an input whose value is based on the assumed maximum vote switch percentage per precinct.

In this paper we will define a sensible attack based on the attacker's driving motivations and fears. The audit plan that results from this analysis is robust and effective against conceivable wholesale attacks that have the potential to reverse an election.

[1] The smallest number of precincts that can produce a fraudulent victory can be calculated and used for this number, See "Designing Mandatory Election Audits", by Jerry Lobdill 8/15/06 or "Random Auditing of E-Voting Systems: How Much Is Enough?", by Howard Stanislevic, 8/16/06, p 6. Of course, this assumes an upper limit on the fraction of votes the attacker is willing to risk switching from one candidate to another in any precinct.

Audit Purpose

The overriding purpose of an audit is to detect and discourage the large-scale wholesale attacks that have been made possible by electronic voting machines, especially the machines currently in use (2006). It is crucial to detect and thwart wholesale attacks that can be implemented by a very small number of people and that can affect the statewide outcome of a federal election.

Some researchers have expressed a desire to uncover all election irregularities, whether caused by deliberate attacks or by software errors or other anomalies. Some such irregularities will tend to produce such bizarre results that inspection will reveal their presence—such as the Tarrant County, Texas March 2006 primary, where the tallying software announced a total vote count of about three times the number of actual voters. Others will produce micro effects such as the corruption of a single DRE or precinct, producing an overall effect that would not change the winner of a race. Detecting small anomalies that cannot change the outcome of a race is not considered to be a purpose of the audit, although if it turns out that this is a frequent result of audits, it will enhance public perception that the audit process provides excellent protection.

The Attacker's Goal, Foreknowledge, and Limitations

We postulate a serious attacker who desperately (but not too desperately) wants her candidate to win. She is not playing hacker games. She will not use her access to attack one or a few precincts or a number of precincts chosen at random. She will not try to reverse an election in a jurisdiction that has historically voted heavily against her wishes, because if successful, she fears that her attack would ring alarm bells and motivate an audit.

She fears that her prediction of the margin against her will be too small, and if so, she will fail in her attempt. Therefore she will switch as many votes as she thinks she can get away with, but she will not risk switching all votes in a precinct to her desired winner, nor will she risk switching more than some estimated maximum percentage of the votes in any precinct, county, or district.

There is clearly a dichotomy between what her desires and her fears tell her to do.

What does she know in advance of the election? She has historic data on voting patterns down to the precinct level. She has a political strategist's estimate of the expected turnout, the direction of the political winds, and an insider's view of how the voting equipment is prepared and the details of the security safeguards in place. She has access to election equipment at the level required to implant a software Trojan Horse in every voting machine and ballot scanner in a county.

Attacker's Trojan Horse

The attacker's Trojan Horse is a security-conscious autonomously operating software program that cannot be detected through testing. It attacks vote counts, not individual ballots. If there is a voter-verified paper trail or paper ballots, an audit will reveal the fraud created by the Trojan Horse. Therefore, the attacker attempts to set the Trojan Horse parameters so that her candidate will win, but no recount will be ordered, and the mandatory election audit has a minimal probability of discovering the fraud. The Trojan Horse operates on the precinct vote count for a particular race.

Pseudocode for the Trojan Horse vote switching algorithm

Calibration Inputs:

Maximum total precinct vote count to attack, L_2.
Minimum total precinct vote count to attack, L_1.
Minimum precinct vote count for the desired loser required to attack, V_{Lmin}.
Fraction of total precinct vote count to switch, V_S.

For each precinct--

At the close of polls read the reported precinct vote count tallies for the desired loser and desired winner, respectively, V_L, and V_W.
Compute the total vote count, $V_T = V_L+V_W$.
If $V_L <= V_{Lmin}$ END
If $V_T > L_2$ END
If $V_T < L_1$ END
If $V_L - V_S \times V_T < 0$,
    $V_L = V_L$
    $V_W = V_W$
    END
else
    $V_L = V_L - V_S \times V_T$
    $V_W = V_W + V_S \times V_T$
END

If L_2 is greater than or equal to the largest vote count in the county the attacker is attacking all the largest precincts. If not, the attacker is trying to fool an audit plan that presumes she will attack all the large precincts.

L_1 is used to avoid corrupting lower vote count precincts and to minimize the number of corrupted precincts in the hope that the audit will miss the precincts that were corrupted.

V_{Lmin} is used to avoid showing a zero vote count for the desired loser unless that situation actually occurred.

V_S is the assumed fraction of the total precinct vote count the attacker believes can be switched without raising suspicion sufficiently to cause a recount. This value depends on

the attacker's desires and fears. Various researchers have assumed values between 0.05 and 0.2.

Excel equations:

New V_L = IF(V_T >Upper_limit, V_L, IF(V_T >Threshold, ROUND(IF(V_L - V_T *Switch_Fraction<Min_New_Loser_VC, V_L, V_L - V_T *Switch_Fraction),0), V_L))

New V_W = IF(New V_L = V_L, V_W, V_W +ROUND(V_T *Switch_Fraction,0))

Sampling Plan

The sampling plan presented below permits the audit designer[2] to make some judgments about how the attacker's dilemma was resolved. If the designer is not willing to make any such judgments then the only safe recourse is to do a full recount. If the audit designer is only willing to assume a value for V_S (vote switch fraction) the design plan reverts to a simple procedure that yields the sample size that would be appropriate if the attacker attacked precincts at random without regard to precinct vote count or attacker plans. This results in over-sampling.

The sampling plan given below finds the smallest number of precincts, N_c, that would reverse the election if corrupted by a specified switching fraction, V_S, of the total precinct count of each precinct attacked and prescribes an order for the audit that minimizes the number of sample precincts actually audited before existing tampering is detected. N_c is the assumed number of corrupt precincts to input into the Poisson formula for sample size, s, along with N, the total number of precincts in the race. The s sample precincts are chosen at random from the total population, N. The sample precincts are then sorted by total vote count, descending. This brings any captured corrupt precincts to the top of the list. Precincts are audited in this order.

Pseudocode for the audit processing of election precinct data is given in the following steps.

1. Sort the precinct returns for the race in question by total vote counts, descending.
2. Compute the winner's total vote margin, M = V_W - V_L
3. Compute the minimum number of votes that must be switched to produce a reversal of the election, (M/2+1).
4. Add four columns to the spreadsheet of election data, (1) the new vote count (after switching votes) for the original loser, (2) the new vote count for the original winner, (3) the new precinct margin for the original loser (4) the running sum of these new precinct margins.

[2] Perhaps these judgments would be better made at the Secretary of State level and applied uniformly in all counties.

5. Starting in the sorted list at the precinct whose vote count is just less than L_2 begin the running sum, and carry it down to the precinct whose vote count just exceeds L_1.
6. Count the number of precincts required for the running sum to just exceed M/2+1. Call this number N_c.
7. Count the number of precincts in the sorted list whose vote count is less than L_2 but greater than L_1. Call this number N.
8. Using a population of N with a corrupt count of N_c calculate the number of precincts, s, to audit for a 0.99 probability of detecting a corrupt precinct.
9. Select the s precincts to audit at random from the population of N precincts.
10. Sort the s precincts in the sample by total vote count, descending. This will bring the most likely corrupted precincts to the top of the list
11. Audit the precincts in sorted order. The audit is complete when a corrupt precinct is found or when all s precincts have been audited.

Landslide elections

When the margin in an election is large and the distribution of precinct vote count sizes has no substantial tail on the high end, reversing the election by switching a reasonable percentage of votes per precinct may require a large number of precincts. In cases like this the calculated sample size may be less that the number of assumed corrupted precincts, or it may be that a reversal cannot be obtained without assuming an unreasonably high percentage of switched votes. To account for such situations it will be necessary to adopt some policy.

It may be thought that all elections should have some minimum amount of auditing regardless of the outcome. There is no mathematically defensible reason for this position, and it seems highly likely that effort expended in such auditing activity will largely be wasted. However, there is no clear, definitive boundary between situations that demand an audit and those where tampering is so unlikely that auditing seems like a silly idea.

In such cases the Poisson equation will still yield a prescribed sample size that matches the mathematical constraints imposed by the audit parameters. The sample size decreases as margin increases. There is no reason not to use this resulting sample size for the audit rather than to adopt some arbitrary floor on audit size—especially if such a floor is proposed to be prescribed in a law whose language does not specifically define the rest of the protocol described here.

It is extremely important to avoid legal language that gives election officials the power to emasculate the mandatory audit process. For example, acceptable legal language might read: The audit sample size shall be the maximum of (a) the sample size computed as described in the previous section, above, or (b) X% of the number of precincts participating in the election in question.

Some final considerations

Perhaps this is a good place to promote the idea that candidates and their campaign staffs probably know better than most others what precincts are suspicious as potential tampering targets. I believe it would be a good idea to allow losers the opportunity to select a single precinct to be included in the audit. This promotes confidence in the election process if nothing else, and in the present situation this would be a definite plus. Roy G. Saltman also proposed this idea in his 1975 report[3].

The protocol in this paper has been tested using real election data from Multnomah County, OR, 2004 presidential election, OH CD 15, 2004, and the Tarrant County, TX 2006 Democratic Party primary election for County Chair. It has also been tested against a variety of vote count distribution possibilities and vote margins with synthetic election data generated using standard simulation modeling techniques.[4]

[3] "Effective Use of Computing Technology in Vote-Tallying", March 1975, National Bureau of Standards, NBSIR 75-687

[4] Simulation Modeling and Analysis, 2nd Edition, by Averill M. Law and W. David Kelton, McGraw-Hill, 1991